Express Mail Label No. EL361715640US

UTILITY PATENT APPLICATION TRANSMITTAL
(Large Entity)

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Docket No.
2204/193

Total Pages in this Submission
62

TO THE ASSISTANT COMMISSIONER FOR PATENTS
Box Patent Application
Washington, D.C. 20231

Transmitted herewith for filing under 35 U.S.C. 111(a) and 37 C.F.R. 1.53(b) is a new utility patent application for an invention entitled:

MULTICAST SWITCHING IN A DISTRIBUTED COMMUNICATION SYSTEM

and invented by:

Da-Hai Ding

If a CONTINUATION APPLICATION, *check appropriate box and supply the requisite information:*

☐ **Continuation** ☐ **Divisional** ☐ **Continuation-in-part (CIP)** of prior application No.: _____

Which is a:

☐ **Continuation** ☐ **Divisional** ☐ **Continuation-in-part (CIP)** of prior application No.: _____

Which is a:

☐ **Continuation** ☐ **Divisional** ☐ **Continuation-in-part (CIP)** of prior application No.: _____

Enclosed are:

Application Elements

1. ☐ Filing fee as calculated and transmitted as described below
2. ☒ Specification having 37 pages and including the following:
   a. ☒ Descriptive Title of the Invention
   b. ☐ Cross References to Related Applications *(if applicable)*
   c. ☐ Statement Regarding Federally-sponsored Research/Development *(if applicable)*
   d. ☐ Reference to Microfiche Appendix *(if applicable)*
   e. ☒ Background of the Invention
   f. ☒ Brief Summary of the Invention
   g. ☒ Brief Description of the Drawings *(if drawings filed)*
   h. ☒ Detailed Description
   i. ☒ Claim(s) as Classified Below
   j. ☒ Abstract of the Disclosure

UTILITY PATENT APPLICATION TRANSMITTAL
(Large Entity)

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Docket No.
2204/193

Total Pages in this Submission
62

Application Elements (Continued)

3. ☒ Drawing(s) *(when necessary as prescribed by 35 USC 113)*

   a. ☐ Formal Number of Sheets _____

   b. ☒ Informal Number of Sheets 17

4. ☒ Oath or Declaration

   a. ☐ Newly executed *(original or copy)* ☒ Unexecuted

   b. ☐ Copy from a prior application (37 CFR 1.63(d)) *(for continuation/divisional application only)*

   c. ☒ With Power of Attorney ☐ Without Power of Attorney

   d. ☐ *DELETION OF INVENTOR(S)*
   Signed statement attached deleting inventor(s) named in the prior application, see 37 C.F.R. 1.63(d)(2) and 1.33(b).

5. ☐ Incorporation By Reference *(usable if Box 4b is checked)*
The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.

6. ☐ Computer Program in Microfiche *(Appendix)*

7. ☐ Nucleotide and/or Amino Acid Sequence Submission *(if applicable, all must be included)*

   a. ☐ Paper Copy

   b. ☐ Computer Readable Copy *(identical to computer copy)*

   c. ☐ Statement Verifying Identical Paper and Computer Readable Copy

Accompanying Application Parts

8. ☐ Assignment Papers *(cover sheet & document(s))*

9. ☐ 37 CFR 3.73(B) Statement *(when there is an assignee)*

10. ☐ English Translation Document *(if applicable)*

11. ☐ Information Disclosure Statement/PTO-1449 ☐ Copies of IDS Citations

12. ☐ Preliminary Amendment

13. ☒ Acknowledgment postcard

14. ☒ Certificate of Mailing

    ☐ First Class ☒ Express Mail *(Specify Label No.):* EL361715640US

UTILITY PATENT APPLICATION TRANSMITTAL
(Large Entity)

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Docket No.
2204/193
Total Pages in this Submission
62

Accompanying Application Parts (Continued)

15. ☐ Certified Copy of Priority Document(s) *(if foreign priority is claimed)*

16. ☐ Additional Enclosures *(please identify below):*

Fee Calculation and Transmittal

CLAIMS AS FILED

For	#Filed	#Allowed	#Extra		Rate	Fee
Total Claims	36	- 20 =	16	x	$18.00	$288.00
Indep. Claims	10	- 3 =	7	x	$78.00	$546.00
Multiple Dependent Claims (check if applicable) ☐						$0.00
					BASIC FEE	$760.00
OTHER FEE *(specify purpose)* _____						$0.00
					TOTAL FILING FEE	$1,594.00

☐ A check in the amount of to cover the filing fee is enclosed.

☐ The Commissioner is hereby authorized to charge and credit Deposit Account No. as described below. A duplicate copy of this sheet is enclosed.

- ☐ Charge the amount of as filing fee.
- ☐ Credit any overpayment.
- ☐ Charge any additional filing fees required under 37 C.F.R. 1.16 and 1.17.
- ☐ Charge the issue fee set in 37 C.F.R. 1.18 at the mailing of the Notice of Allowance, pursuant to 37 C.F.R. 1.311(b).

Jeffrey T. Klayman
Signature

Dated: October 13, 1999

Jeffrey T. Klayman
BROMBERG & SUNSTEIN LLP
125 Summer Street
Boston, MA 02110
(617) 443-9292

cc:

CERTIFICATE OF MAILING BY "EXPRESS MAIL" (37 CFR 1.10) Applicant(s): Ding			Docket No. 2204/193
Serial No. Not Yet Assigned	Filing Date Herewith	Examiner Not Yet Assigned	Group Art Unit Not Yet Assigned

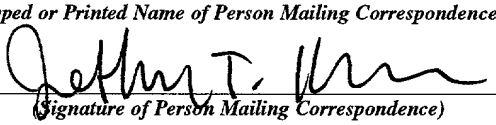
Invention: MULTICAST SWITCHING IN A DISTRIBUTED COMMUNICATION SYSTEM

I hereby certify that this Utility Patent Application Transmittal and enclosures referred to therein
(Identify type of correspondence)

is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 in an envelope addressed to: The Assistant Commissioner for Patents, Washington, D.C. 20231 on

October 13, 1999
(Date)

Jeffrey T. Klayman
(Typed or Printed Name of Person Mailing Correspondence)


(Signature of Person Mailing Correspondence)

EL361715640US
("Express Mail" Mailing Label Number)

Note: Each paper must have its own certificate of mailing.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR UNITED STATES PATENT

FOR

MULTICAST SWITCHING IN A DISTRIBUTED COMMUNICATION SYSTEM

Inventors:

Da-Hai Ding
10 Baker Avenue
Lexington, MA 02421

Attorney Docket: 2204/193 (BA446)

Attorneys:

BROMBERG & SUNSTEIN LLP
125 Summer Street
Boston, MA 02110
(617) 443-9292

MULTICAST SWITCHING IN A DISTRIBUTED COMMUNICATION SYSTEM

CROSS-REFERENCE TO RELATED APPLICATIONS

FIELD OF THE INVENTION

The present invention relates generally to communication systems, and more particularly to multicast switching in a distributed communication system.

BACKGROUND OF THE INVENTION

In today's information age, it is typical for computers and computer peripherals to be internetworked over a communication network. The communication network typically includes a plurality of communication links that are interconnected through a number of intermediate devices, such as bridges, routers, or switches. Information sent by a source device to a destination device traverses one or more communication links.

The various communication devices in the communication network, including the computers, computer peripherals, and intermediate devices, utilize various communication protocols in order to transport the information from the source device to the destination device. The communication protocols are typically implemented in layers, which together form a protocol stack. Each protocol layer provides a specific set of services to the protocol layer immediately above it in the protocol stack. Although there are different protocol layering schemes in use today, the different protocol layering schemes have certain common attributes. Specifically, protocols at the lowest layer in the protocol stack, which are typically referred to as the "layer 1" or "physical layer" protocols, define the physical and electrical characteristics for transporting the information from one communication device to another communication device across a single communication link. Protocols at the next layer in the protocol stack, which are typically referred to as the "layer 2" or "Medium Access Control (MAC) layer" protocols, define the protocol

message formats for transporting the information across the single communication link by the physical layer protocols. Protocols at the next layer in the protocol stack, which are typically referred to as the "layer 3" or "network layer" protocols, define the protocol message formats for transporting the information end-to-end from the source device to the destination device across multiple communication links. Higher layer protocols ultimately utilize the services provided by the network protocols for transferring information across the communication network.

Each intermediate device supports multiple communication links, and forwards protocol messages between communication links. Some intermediate devices forward protocol messages based upon layer 2 addresses, while other intermediate devices forward protocol messages based upon layer 3 addresses. An intermediate device that forwards protocol messages based upon layer 2 addresses is often referred to as a bridge or switch, while an intermediate device that forwards protocol messages based upon layer 3 addresses is often referred to as a router.

One common networking application, referred to as "multicasting," allows an information provider (referred to hereinafter as a "multicast source") to transmit a single unit of multicast information (referred to hereinafter as a "multicast packet") simultaneously to a group of information consumers (referred to hereinafter as the "multicast group"), specifically by addressing the multicast packet to the multicast group using a multicast address. The multicast group members monitor the communication network for multicast packets addressed to the multicast group.

In a communication network that supports Internet Protocol (IP) multicasting, multicasting is supported at layer 3 of the protocol stack. The multicast address is a layer 3 address, and the various routers in the communication network forward multicast packets using the layer 3 address according to a multicast routing protocol.

When a layer 2 device needs to forward a multicast packet, the layer 2 device may treat the multicast packet as a broadcast, in which case the layer 2 device forwards the multicast packet to all communication links that it supports (other than the communication link over which the multicast packet was received). Such broadcasting of the multicast packet may generate unnecessary network traffic, particularly when multicast streams are

not supported by all communication links. Therefore, the layer 2 device preferably determines which communication links support multicast streams, and selectively forwards the multicast packet only to those communication links that support multicast streams. Such selective forwarding of the multicast packet avoids unnecessary network traffic.

One way for the layer 2 device to determine which communication links support multicast streams is by monitoring Internet Group Management Protocol (IGMP) messages received from each of the communication links. IGMP is a protocol that is used by multicast devices to control membership in multicast groups. Using IGMP, a multicast device can join a multicast group or leave a multicast group. IGMP messages are layer 3 protocol messages that identify multicast devices (i.e., hosts and routers) on the communication link. Even though the IGMP messages are layer 3 messages, the layer 2 device is able to receive and process the IGMP messages in order to determine whether or not there are any active multicast devices on a particular communication link. Such monitoring of IGMP messages is referred to hereinafter as IGMP "snooping."

To further reduce unnecessary network traffic, the layer 2 device may forward a single IGMP report on behalf of all connected multicast hosts for a particular multicast group and suppress any additional IGMP reports for the same multicast group. When a multicast device generates an IGMP query for the multicast group, the layer 2 device may receive IGMP reports from all connected multicast hosts. Since only one IGMP report is needed to indicate that the multicast group is active (i.e., that the multicast group has at least one member), the layer 2 device forwards only one of the IGMP reports and suppresses the other IGMP reports. Such forwarding and suppressing of IGMP reports is referred to hereinafter as IGMP "proxy."

Because layer 2 devices are utilized in a wide range of applications, some layer 2 devices utilize a modular design that enables a number of modules to be interconnected in a stack configuration such that the number of interconnected modules interoperate in a cooperating mode of operation to form a single virtual device. Each module is capable of operating independently as a stand-alone device or in a stand-alone mode of operation, and therefore each module is a complete system unto itself. Each module typically includes a

number of network interfaces for supporting directly connected communication links as well as an inter-module communication interface for interfacing with the other modules in the stack.

The modular design approach enables the layer 2 device to be scalable, such that modules can be added and removed to fit the requirements of a particular application. However, the modular design approach also complicates IGMP snooping and IGMP proxy functions. Specifically, when a number of modules are interconnected in a cooperating mode of operation, it is possible for multicast group members on different communication links to be supported by different modules of the layer 2 device.

SUMMARY OF THE INVENTION

In accordance with one aspect of the invention, multicast switching is supported in a distributed (modular) communication system having a plurality of cooperating modules by having each module forward multicast packets to all remote modules that are associated with a particular multicast stream. No centralized module or logic is required. The same logic is used in a particular module whether the module is operating in a cooperating mode of operation or in a stand-alone mode of operation. A multicast stream can be associated with a particular multicast group or with a particular multicast group on a particular Virtual LAN (VLAN). Each module determines the network interfaces and the remote modules that are associated with a particular multicast stream. When the module receives a multicast packet from a network interface, the module forwards the multicast packet to all network interfaces that are associated with the multicast stream other than the network interface over which the multicast packet was received, and also forwards the multicast packet to all remote modules that are associated with the multicast stream. When the module receives a multicast packet from a remote module, the module forwards the multicast packet to all network interfaces that are associated with the multicast stream.

In accordance with another aspect of the invention, the modules use IGMP snooping to determine the network interfaces and remote modules that are associated with each multicast stream. A module determines that a network interface is associated with a

multicast stream upon receiving an IGMP message over the network interface. A module determines that a remote module is associated with a multicast stream upon receiving an IGMP message from the remote module. Therefore, when a module receives an IGMP message over a network interface, the module forwards the IGMP message to all remote modules, among other things.

In accordance with another aspect of the invention, each module maintains a forwarding interface list for each multicast stream. The forwarding interface list is preferably a bit-mapped field with one bit per network interface and one bit per remote module. The forwarding interface list indicates all network interfaces and all remote modules that are associated with the corresponding multicast stream. The forwarding interface list is preferably programmed into a CAM device that is used for switching packets.

In accordance with another aspect of the invention, an intermediate device sends IGMP messages using a multicast device address that is learned from a received IGMP message. An IGMP report message is sent using a multicast host address. An IGMP query message is sent using a multicast router address.

In acordance with another aspect of the invention, an intermediate device sends IGMP messages over an interface, and formats the IGMP messages as either IGMP version 1 messages or IGMP version 2 messages according to the IGMP version of IGMP messages received over the interface. The intermediate device sends IGMP version 1 messages over the interface if at least one IGMP version 1 message was received over the interface. The intermediate device sends IGMP version 2 messages over the interface if and only if all IGMP messages received over the interface are IGMP version 2 messages.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing and other objects and advantages of the invention will be appreciated more fully from the following further description thereof with reference to the accompanying drawings wherein:

FIG. 1 is a block diagram showing an exemplary modular Ethernet switch in

accordance with a preferred embodiment of the present invention;

FIG. 2 is a block diagram showing the relevant logic blocks of an exemplary module in accordance with an embodiment of the present invention;

FIG. 3 is a logic flow diagram showing exemplary logic for forwarding a multicast packet received over a network port in accordance with an embodiment of the present invention;

FIG. 4 is a logic flow diagram showing exemplary logic for forwarding a multicast packet received from a remote module over the inter-module communication interface in accordance with an embodiment of the present invention;

FIG. 5 is a logic flow diagram showing exemplary logic for forwarding a multicast packet received over a network port in accordance with an embodiment of the present invention;

FIG. 6 is a logic flow diagram showing exemplary logic for forwarding a multicast packet received from a remote module over the inter-module communication interface in accordance with an embodiment of the present invention;

FIG. 7 shows the relevant information that a module maintains for each multicast stream in accordance with an embodiment of the present invention;

FIG. 8 is a logic flow diagram showing exemplary logic for enabling IGMP snooping in accordance with a preferred embodiment of the present invention;

FIG. 9 is a logic flow diagram showing exemplary logic for processing an IGMP query message received over a network interface in accordance with a preferred embodiment of the present invention;

FIG. 10 is a logic flow diagram showing exemplary logic for processing an IGMP report message received over a network interface in accordance with a preferred embodiment of the present invention;

FIG. 11 is a logic flow diagram showing exemplary logic for processing an IGMP query message received from a remote module in accordance with a preferred embodiment of the present invention;

FIG. 12 is a logic flow diagram showing exemplary logic for processing an IGMP report message received from a remote module in accordance with a preferred embodiment

of the present invention;

FIG. 13 is a logic flow diagram showing exemplary logic for sending an IGMP report message as part of the IGMP proxy function in accordance with a preferred embodiment of the present invention;

FIG. 14 is a logic flow diagram showing exemplary logic for sending an IGMP query message in accordance with a preferred embodiment of the present invention;

FIG. 15 is a logic flow diagram showing exemplary logic for forwarding a multicast packet in accordance with a preferred embodiment of the present invention;

FIG. 16 is a logic flow diagram showing exemplary logic for removing an inactive remote module in accordance with a preferred embodiment of the present invention; and

FIG. 17 is a logic flow diagram showing exemplary logic for entering a stand-alone mode of operation in accordance with a preferred embodiment of the present invention.

DETAILED DESCRIPTION OF A PREFERRED EMBODIMENT

An embodiment of the present invention supports multicast switching in a distributed (modular) communication system without requiring a dedicated management module. The distributed (modular) communication system includes multiple interconnected modules. Each module includes a number of network interfaces and an inter-module communication interface. Each network interface is coupled to a Local Area Network (LAN), and is preferably capable of supporting multiple Virtual LANs (VLANs) over the LAN. The inter-module communication interface enables the module to communicate with the other modules. Each module has a logical connection to each of the other cooperating modules over the inter-module communication interface.

FIG. 1 is a block diagram showing an exemplary distributed (modular) communication system 100 in accordance with an embodiment of the present invention. The distributed (modular) communication system 100 includes a number of modules (110, 120) that are interconnected over a communication medium 140, such as a bus, backplane, or other signaling medium. For convenience, the distributed (modular) communication

system 100 is referred to hereinafter as a stack.

FIG. 2 is a block diagram showing the relevant logic blocks of an exemplary module 110 in accordance with an embodiment of the present invention. The module 110 includes, among other things, a number of network interfaces (113, 114), an inter-module communication interface (116), switching logic (112), IGMP snooping/proxy logic (115), and an IGMP snooping/proxy database (111).

In a preferred embodiment of the invention, the stack 100 is a modular Ethernet switch including a number of interconnected Ethernet switching modules. The preferred Ethernet switching module can be configured to operate as an independent stand-alone device, or alternatively up to eight (8) Ethernet switching modules can be interconnected in a stack configuration, preferably by interconnecting the up to eight (8) Ethernet switching modules through a dual ring bus having a bandwidth of 2.5 gigabits per second. Within the stack configuration, a particular Ethernet switching module can be configured to operate in either a stand-alone mode, in which the particular Ethernet switching module performs Ethernet switching independently of the other Ethernet switching modules in the stack, or a cooperating mode, in which the particular Ethernet switching module performs Ethernet switching in conjunction with other cooperating Ethernet switching modules. Furthermore, a particular Ethernet switching module in the stack can be dynamically reconfigured between the stand-alone mode and the cooperating mode without performing a system reset or power cycle of the particular Ethernet switching module, and Ethernet switching modules can be dynamically added to the stack and removed from the stack without performing a system reset or power cycle of the other Ethernet switching modules in the stack.

In a preferred embodiment, each network interface (113, 114) is attached to an Ethernet Local Area Network (LAN) on which there are a number of directly connected communication devices (not shown in FIG. 1). Each network interface (113, 114) may be configured to support virtual LANs (VLANs) over the Ethernet LAN. Thus, each directly connected communication device is associated with a particular network interface on a particular Ethernet switching module, and, if VLAN is supported, with a particular VLAN associated with the network interface.

In a preferred embodiment, the inter-module communication interface (116) is coupled to the dual ring bus 140, and enables the corresponding Ethernet switching module to communicate with the other Ethernet switching modules in the stack 100. The inter-module communication interface (116) enables the module 110 to send and receive information (including management information, control information, and data) over the dual-ring bus 140. The inter-module communication interface (116) provides for both reliable (acknowledged) and unreliable transfers over the dual-ring bus 140, and allows information to be directed to a particular remote module or group of remote modules. For convenience, a virtual connection from the module 110 to a remote module over the inter-module communication interface (116) is referred to hereinafter as a "stack port." The module 110 includes a stack port for each remote module.

The IGMP snooping/proxy logic (115) performs the IGMP snooping and IGMP proxy functions. The IGMP snooping/proxy logic (115) includes, among other things, logic for determining the last host address for each multicast stream, logic for determining the last router address for each multicast stream, logic for determining the interfaces associated with each multicast stream, logic for determining the IGMP version for each interface, and logic for consolidating multiple IGMP reports into a single IGMP report as part of the IGMP proxy function. The IGMP snooping/proxy logic (115) stores certain information in the IGMP snooping/proxy database (111). The IGMP snooping/proxy logic (115) preferably also configures a forwarding interface list (not shown) that is used to by the switching logic (112) for forwarding packets among and between the network interfaces (113, 114) and the remote modules (stack ports) via the inter-module communication interface (116). The forwarding interface list indicates all network interfaces and all remote modules (stack ports) that are associated with the corresponding multicast stream.

The switching logic (112) forwards protocol messages among and between the network interfaces (113, 114), the remote modules (stack ports), and the IGMP snooping/proxy logic (115). Among other things, the switching logic (112) utilizes the forwarding interface list to forward multicast packets among and between the network interfaces (113, 114) and the remote modules (stack ports), and also forwards IGMP

messages to the IGMP snooping/proxy logic (115).

In a preferred embodiment of the invention, each module can be configured to operate as an independent stand-alone device or in a stand-alone mode within the stack. Thus, each module includes management/control logic (not shown) that enables the module to be individually managed and controlled, for example, through a console user interface, a Simple Network Management Protocol (SNMP) session, or a world wide web session. Therefore, the preferred management/control logic includes, among other things, a Transmission Control Protocol/Internet Protocol (TCP/IP) stack, an SNMP agent, and a web engine. Furthermore, each module is assigned MAC and IP addresses, allowing each module to send and receive management and control information independently of the other modules. When operating in a cooperating mode of operation, each cooperating module uses the same MAC and IP addresses for management and control, so that the cooperating modules appear and operate as a single device having a single MAC/IP address pair. In a preferred embodiment, all cooperating modules use the MAC and IP addresses of a base module. The base module is described below.

The management/control logic maintains a number of management databases (not shown) for storing configuration and operational information. The management/control logic maintains a management database containing network management objects and parameters that are related to a particular port or interface, and maintains another management database containing network management objects and parameters that are system-wide in scope. When the module is operating in a cooperating mode within the stack, the management database containing network management objects and parameters that are system-wide in scope is referred to as the "shadowed" management database, and the management database containing network management objects and parameters that are related to a particular port or interface is referred to as the "segmented" management database.

In order for multiple modules to operate in a cooperating mode of operation, one of the modules is configured as a base module for the stack, specifically by a user-controlled toggle switch on the module. Any module can be configured as the base module. The base module is not a dedicated management module for the entire stack, and does not

require any additional hardward or software. The base module is responsible for ensuring orderly stack formation, handling stack configuration changes such as module insertion and removal, and periodically verifying stack integrity. The base module configuration is used if a configuration conflict is identified during stack formation.

Prior to forming a stack or allowing a new module to join a stack, all modules exchange their configuration databases with the other modules. Management objects relating to IGMP snooping and IGMP proxy include, among other things, a robustness value and a query time. Management objects relating to VLAN include, among other things, VLAN-based static port masks. A database exchange event is used by a module to broadcast its configuration databases to all modules in the stack. Received databases are cached in a temporary buffer. The database exchange event does not alter a module's operating mode.

In case of a database exchange failure, a discard temporary database event is sent to all modules. Upon receiving the discard temporary database event, each module discards the configuration databases that are cached in the temporary buffer. The discard temporary database event does not alter a module's operating mode.

Once all modules have exchanged their configuration databases, a join stack event is used to form the stack. At this time, user-specified configuration data are synchronized, and any configuration conflicts are resolved across the stack. Also, various processes, including IGMP snooping and IGMP proxy processes, are synchronized across all modules of the stack so that the stack operates as a single device.

In order to perform multicast switching across the multiple interconnected modules, each module determines the interfaces that are associated with each multicast stream, and forwards each multicast packet over the interfaces that are associated with the corresponding multicast stream. A multicast stream corresponds to a particular multicast group, and, if VLAN is enabled, to a particular multicast group on a particular VLAN. It should be noted that a particular multicast group can be supported on multiple VLANs, even on the same interface.

When multicast switching is performed using broadcasting, all interfaces are considered to be associated with the multicast stream. Specifically, all network interfaces

and all remote modules (stack ports) are considered to be associated with the multicast stream. When a multicast packet is received over a network interface, the multicast packet is forwarded to the other network interfaces, and is also forwarded to the remote modules via the inter-module communication interface. When a multicast packet is received from a remote module over the inter-module communication interface, the multicast packet is forwarded to all network interfaces.

FIG. 3 is a logic flow diagram showing exemplary logic 300 for forwarding a multicast packet received over a network interface in accordance with an embodiment of the present invention. Beginning at step 302, and upon receiving a multicast packet over a network interface, in step 304, forwards the multicast packet to all network interfaces other than the network interface over which the multicast packet was received, in step 306. The logic also forwards the multicast packet to all remote modules (stack ports) via the inter-module communication interface, in step 308. The logic 300 terminates in step 399.

FIG. 4 is a logic flow diagram showing exemplary logic 400 for forwarding a multicast packet received from a remote module over the inter-module communication interface in accordance with an embodiment of the present invention. Beginning at step 402, and upon receiving a multicast packet from a remote module via the inter-module communication interface, in step 404, the logic forwards the multicast packet to all network interfaces, in step 406. The logic 400 terminates in step 499.

When IGMP snooping is enabled, only interfaces that support a multicast device are considered to be associated with the corresponding multicast stream. A network interface is considered to be associated with a particular multicast stream if at least one multicast device associated with the multicast stream is supported on the network interface. A remote module (stack port) is considered to be associated with a particular multicast stream if at least one multicast device associated with the multicast stream is supported on any of the network interfaces supported by the remote module. Thus, a stack port is essentially an aggregation of all network interfaces supported by the corresponding remote module.

FIG. 5 is a logic flow diagram showing exemplary logic 500 for forwarding a multicast packet received over a network port in accordance with an embodiment of the

present invention. Beginning at step 502, and upon receiving a multicast packet over a network interface, in step 504, the logic proceeds to determine the multicast stream that is associated with the multicast packet, in step 506. The logic then determines the network interfaces that are associated with the multicast stream, in step 508, and forwards the multicast packet to all network interfaces that are associated with the multicast stream other than the network interface over which the multicast packet was received, in step 510. The logic also determines the remote modules (stack ports) that are associated with the multicast stream, in step 512, and forwards the multicast packet to all remote modules (stack ports) that are associated with the multicast stream via the inter-module communication interface (116), in step 514. The logic 500 terminates in step 599.

FIG. 6 is a logic flow diagram showing exemplary logic 600 for forwarding a multicast packet received from a remote module over the inter-module communication interface (116) in accordance with an embodiment of the present invention. Beginning at step 602, and upon receiving a multicast packet from a remote module over the inter-module communication interface (116), in step 604, the logic proceeds to determine the multicast stream that is associated with the multicast packet, in step 606. The logic then determines the network interfaces that are associated with the multicast stream, in step 608, and forwards the multicast packet to all network interfaces that are associated with the multicast stream, in step 610. The logic 600 terminates in step 699.

There are various ways in which to determine the interfaces that are associated with the multicast stream. For example, IGMP snooping can be used to automatically determine the interfaces based upon IGMP messages that are received over the various interfaces. Alternatively, or additionally, a network manager can configure the interfaces that are associated with the multicast stream via the management/control logic. This latter approach may be useful, for example, to configure interfaces that do not support IGMP.

IGMP snooping in a distributed (modular) communication system requires that all IGMP messages be distributed to all modules. Therefore, when a module receives an IGMP message over a network interface, the module forwards the IGMP message to all remote modules via the inter-module communication interface (116).

Each module determines the interfaces (i.e., the network interfaces and stack ports)

that are associated with a particular multicast stream based upon IGMP messages received over the interfaces. In a preferred embodiment of the invention, each module maintains a forwarding interface list for each multicast stream. Each forwarding interface list indicates the interfaces that are associated with the corresponding multicast stream. The switching logic 204 utilizes the forwarding interface list(s) to forward multicast packets among and between the network interfaces and the remote modules (stack ports).

In addition to determining that a particular interface is associated with the multicast stream, the module is able to determine whether the interface supports a multicast host device and/or a multicast router. An IGMP query message received over an interface indicates that the interface supports a multicast router, since multicast routers generate IGMP query messages. An IGMP report message received over an interface indicates that the interface supports a multicast host device, since multicast host devices generated IGMP report messages. In a preferred embodiment of the invention, each module maintains a host interface list and a router interface list for each multicast stream. The host interface list indicates the network interfaces and remote modules (stack ports) that support multicast host devices associated with the corresponding multicast stream. The router interface list indicates the network interfaces and remote modules (stack ports) that support multicast routers associated with the corresponding multicast stream. An interface is considered to be associated with the multicast stream, and is therefore included in the forwarding interface list, if the interface is in either the host interface list or the router interface list.

Although a module is neither a multicast host device nor a multicast router, it is sometimes necessary or desirable for a module to generate IGMP query messages and/or IGMP report messages in order to support IGMP snooping and IGMP proxy in a distributed (modular) communication system. The module may send an IGMP query message, for example, to verify that a particular interface supports a multicast stream before forwarding multicast packets to the interface or to verify that a particular interface no longer supports a multicast stream before "pruning" the interface, specifically as part of the IGMP snooping function. The module may send an IGMP report message, for example, to report the multicast stream(s) supported by the module in response to an

IGMP query message, specifically as part of the IGMP proxy function.

As with other protocol messages, the IGMP messages require a source MAC address and a source IP address. In a prior art embodiment, the module utilizes its own MAC and IP addresses for sending IGMP messages. This makes the module visible to other layer 3 devices, which is undesirable. Also, if the module supports multiple VLANs, then the module would need to support a separate MAC/IP address pair for each VLAN, which is an inefficient use of addresses.

Therefore, a preferred embodiment utilizes a host address for sending IGMP report messages, and utilizes a router address for sending IGMP query messages. The module learns host addresses and router addresses from received IGMP messages. The module maintains a last host address and a last router address for each multicast stream. The module utilizes the last host address for sending an IGMP report message, and utilizes the last router address for sending an IGMP query message. By using host and router addresses to send IGMP messages, the module remains transparent to layer 3 devices, and does not need to support a separate MAC/IP address pair for each VLAN.

When the module sends an IGMP message on a particular interface, the module must decide whether to format the IGMP message as an IGMP version 1 message or an IGMP version 2 message. In a prior art embodiment, the module may be configured to send IGMP version 1 messages or IGMP version 2 messages on the interface. This precludes the module from operating as a "plug-and-play" device, since the module requires an explicit configuration in order to operate in the communication network.

A preferred embodiment automatically determines an IGMP version for each interfaces based upon received IGMP messages. If the module receives only IGMP version 2 messages over a particular interface, then the module considers the interface to be an IGMP version 2 interface, and the module generates IGMP version 2 messages on the interface. If the module receives at least IGMP version 1 message over the interface, then the module considers the interface to be an IGMP version 1 interface, and the module generates IGMP version 1 messages on the interface. The module maintains an IGMP version indicator for each interface on each multicast stream. By determining the IGMP version based upon received IGMP messages, the module is able to operate as a "plug-

and-play" device.

When the module 110 receives an IGMP query message over a network interface, such as the network interface (113), the switching logic (112) forwards the IGMP query message to all remote modules (stack ports) via the inter-module communication interface (116), and forwards the IGMP query message to all network interfaces other than the network interface over which the IGMP query message was received (such as the physical Ethernet port 114). The switching logic (112) also forwards the IGMP query message to the IGMP snooping/proxy logic (115) for local processing.

When the module 110 receives an IGMP report message over a network interface, such as the network interface (113), the switching logic (112) forwards the IGMP report message to all remote modules (stack ports) via the inter-module communication interface (116), and, if IGMP proxy is disabled, forwards the IGMP report message to all network interfaces other than the network interface over which the IGMP report message was received (such as the network interface 114). The switching logic (112) also forwards the IGMP report message to the IGMP snooping/proxy logic (115) for local processing.

When the module 110 receives an IGMP query message from a remote module over the inter-module communication interface (116), the switching logic forwards the IGMP query message to all network interfaces (113, 114). The switching logic (112) also forwards the IGMP query message to the IGMP snooping/proxy logic (115) for local processing.

When the module 110 receives an IGMP report message from a remote module over the inter-module communication interface (116), the switching logic (112) forwards the IGMP report message to all network interfaces (113, 114) only if IGMP proxy is disabled. The switching logic (112) also forwards the IGMP report message to the IGMP snooping/proxy logic (115) for local processing.

The IGMP snooping/proxy logic (115) determines, among other things, a last host address for each multicast stream, a last router address for each multicast stream, the interfaces associated with each multicast stream, and an IGMP version for each interface based upon IGMP messages forwarded to the IGMP snooping/proxy logic (115) by the switching logic (112). The IGMP snooping/proxy logic (115) determines the last router

address associated with the multicast stream from an IGMP query message. The IGMP snooping/proxy logic (115) determines the last host address associated with the multicast stream from an IGMP report message. The IGMP snooping/proxy logic (115) determines an IGMP version for each interface based upon the format of IGMP messages received over the interface. The IGMP snooping/proxy logic (115) determines whether a particular network interface is associated with a particular multicast stream based upon IGMP messages received over the network interface. The IGMP snooping/proxy logic (115) determines whether a particular remote module (stack port) is associated with a particular multicast stream based upon IGMP messages received from the remote module over the inter-module communication interface (116). The IGMP snooping/proxy logic (115) stores the appropriate information in the IGMP snooping/proxy database (111).

FIG. 7 shows the relevant information that the module maintains in the IGMP snooping/proxy database (111) for each multicast stream. Specifically, for each multicast stream, the module maintains a last host address 702, a last router address 704, a host interface list 706, a router interface list 708, and an IGMP version list 710. The last host address 702 stores the last host address that the module learned for the corresponding multicast stream. The last router address 704 stores the last router address that the module learned for the corresponding multicast stream. The host interface list 706 indicates all interfaces that are associated with the multicast stream and support multicast host devices. The router interface list 708 indicates all interfaces that are associated with the multicast stream and support multicast routers. The IGMP version list 710 indicates the IGMP version for each interface. In a preferred embodiment, the host interface list 706, router interface list 708, and IGMP version list 710 are bit-mapped fields having one bit per interface, that is, one bit for each network interface and one bit for each remote module (stack port). Each bit in the host interface list 706 indicates whether the corresponding interface supports a multicast host device associated with the multicast stream. Each bit in the router interface list 708 indicates whether the corresponding interface supports a multicast router associated with the multicast stream. Each bit in the IGMP version list 710 indicates whether the corresponding interface is an IGMP version 1 interface or an IGMP version 2 interface. The IGMP snooping/proxy logic (115) uses the host interface

list 706 and the router interface list 708 to configure the forwarding interface list.

Thus, upon obtaining an IGMP message from the switching logic (112), the IGMP snooping/proxy logic (115) determines the multicast stream associated with the IGMP message. If the IGMP message is an IGMP query message from a particular router, then the IGMP snooping/proxy logic (115) saves the router address as the last router address 704 associated with the multicast stream, and adds the interface to the router interface list 708 associated with the multicast stream. If the IGMP message is an IGMP report message from a particular multicast host, then the IGMP snooping/proxy logic (115) saves the host address as the last host address 702 associated with the multicast stream, and adds the interface to the host interface list 706 associated with the multicast stream. The IGMP snooping/proxy logic (115) also determines the IGMP version for the IGMP message, and indicates the IGMP version for the interface in the IGMP version list 710. In a preferred embodiment, an interface is considered to be an IGMP version 1 interface if at least one IGMP version 1 message was received over the interface, and is considered to be an IGMP version 2 interface if only IGMP version 2 messages are received over the interface.

Additionally, the IGMP snooping/proxy logic (115) configures the forwarding interface list based upon the host interface list and the router interface list. In a preferred embodiment, the forwarding interface list is a bit-mapped field having one bit per interface that is programmed into a CAM chip that is used by the switching logic (112) to forward packets. A particular interface is included in the forwarding interface list if the interface is included in the host interface list and/or the router interface list.

When the IGMP snooping/proxy logic (115) sends an IGMP query message on a particular interface, for example, as part of the IGMP snooping function, the IGMP snooping/proxy logic (115) uses the last router address associated with the multicast stream, and formats the IGMP query message according to the IGMP version indicator associated with the interface. The IGMP snooping/proxy logic (115) generates two types of IGMP query messages, specifically an "all-aboard" query and a "last member" query. The "all-aboard" query is used to determine all interfaces that are associated with a particular multicast stream when IGMP snooping is first enabled. This prevents the module from accidentally removing an interface that supports a multicast device from the

forwarding interface list, which would disrupt multicast services to the multicast device. The "last member" query is used to ensure that there are no multicast devices on the interface before removing the interface from the forwarding interface list.

When the IGMP snooping/proxy logic (115) sends an IGMP report message on a particular interface, for example, as part of the IGMP proxy function, the IGMP snooping/proxy logic (115) uses the last host address associated with the multicast stream, and formats the IGMP report message according to the IGMP version indicator associated with the interface. The module may send an IGMP report message, for example, to report the multicast stream(s) supported by the module in response to an IGMP query message.

FIG. 8 is a logic flow diagram showing exemplary logic 800 for enabling IGMP snooping in accordance with a preferred embodiment of the present invention. Beginning at step 802, and upon determining that IGMP snooping is enabled for a particular multicast stream, in step 804, the logic waits for at least one IGMP query message associated with the multicast stream. Upon receiving an IGMP query message associated with the multicast stream from a router over an interface, in step 806, the logic saves the router address as the last router address associated with the multicast stream, in step 808, and sends an IGMP "all aboard" query message over all network interfaces using the last router address, in step 810. The logic 800 terminates in step 899.

FIG. 9 is a logic flow diagram showing exemplary logic 900 for processing an IGMP query message received over a network interface in accordance with a preferred embodiment of the present invention. Beginning at step 902, and upon receiving an IGMP query message from a router over a network interface, in step 904, the logic determines the multicast stream associated with the IGMP query message, in step 906, and saves the router address as the last router address associated with the multicast stream, in step 908. The logic then determines the IGMP version for the IGMP query message, in step 910, and, if the IGMP query message is an IGMP version 1 message (YES in step 912), the logic sets the IGMP version for the network interface to IGMP version 1, in step 914. If IGMP snooping is enabled (YES in step 916), then the logic adds the network interface to the router interface list associated with the multicast stream, in step 918. If IGMP proxy is enabled (YES in step 920) and the host interface list is not empty (NO in step 922), then

the logic starts an IGMP report timer, in step 924. In any case, the logic forwards the IGMP query message to all network interfaces other than the network interface over which the IGMP query message was received, in step 926, and forwards the IGMP query message along with a module identifier to all remote modules (stack ports) via the inter-module communication interface, in step 928. The logic 900 terminates in step 999.

FIG. 10 is a logic flow diagram showing exemplary logic 1000 for processing an IGMP report message received over a network interface in accordance with a preferred embodiment of the present invention. Beginning at step 1002, and upon receiving an IGMP report message from a host over a network interface in step 1004, the logic determines the multicast stream associated with the IGMP report message, in step 1006, and saves the host address as the last host address associated with the multicast stream, in step 1008. The logic then determines the IGMP version for the IGMP report message, in step 1010, and, if the IGMP report message is an IGMP version 1 message (YES in step 1012), the logic sets the IGMP version for the network interface to IGMP version 1, in step 1014. If IGMP snooping is enabled (YES in step 1016), then the logic adds the network interface to the host interface list associated with the multicast stream, in step 1018. The logic forwards the IGMP report message along with a module identifier to all remote modules (stack ports) via the inter-module communication interface, in step 1020. If IGMP proxy is disabled (NO in step 1022), then the logic forwards the IGMP report message to all network interfaces other than the network interface over which the IGMP report message was received, in step 1024. The logic 1000 terminates in step 1099.

FIG. 11 is a logic flow diagram showing exemplary logic 1100 for processing an IGMP query message received from a remote module in accordance with a preferred embodiment of the present invention. Beginning at step 1102, and upon receiving an IGMP query message from a router over the inter-module communication interface including a module identifier identifying a remote module, in step 1104, the logic determines the multicast stream associated with the IGMP query message, in step 1106, and saves the router address as the last router address associated with the multicast stream, in step 1108. The logic then determines the IGMP version for the IGMP query message, in step 1110, and, if the IGMP query message is an IGMP version 1 message (YES in step

1112), the logic sets the IGMP version for the remote module (stack port) to IGMP version 1, in step 1114. If IGMP snooping is enabled (YES in step 1116), then the logic adds the remote module (stack port) to the router interface list associated with the multicast stream, in step 1118. In any case, the logic forwards the IGMP query message to all network interfaces, in step 1120. The logic 1100 terminates in step 1199.

FIG. 12 is a logic flow diagram showing exemplary logic 1200 for processing an IGMP report message received from a remote module in accordance with a preferred embodiment of the present invention. Beginning at step 1202, and upon receiving an IGMP report message from a host over the inter-module communication interface including a module identifier identifying a remote module, in step 1204, the logic determines the multicast stream associated with the IGMP report message, in step 1206, and saves the host address as the last host address associated with the multicast stream, in step 1208. The logic then determines the IGMP version for the IGMP report message, in step 1210, and, if the IGMP report message is an IGMP version 1 message (YES in step 1212), the logic sets the IGMP version for the remote module (stack port) to IGMP version 1, in step 1214. If IGMP snooping is enabled (YES in step 1216), then the logic adds the remote module (stack port) to the host interface list associated with the multicast stream, in step 1218. If IGMP proxy is disabled (NO in step 1220), then the logic forwards the IGMP report message to all network interfaces, in step 1222. The logic 1200 terminates in step 1299.

FIG. 13 is a logic flow diagram showing exemplary logic 1300 for sending an IGMP report message as part of the IGMP proxy function in accordance with a preferred embodiment of the present invention. Beginning at step 1302, and upon detecting that the IGMP report timer expired, in step 1304, the logic sends IGMP report messages to the routers using the last host address, in step 1306, specifically by sending the IGMP report messages over the network interfaces from which the IGMP query messages were received. The logic 1300 terminates in step 1399.

FIG. 14 is a logic flow diagram showing exemplary logic 1400 for sending an IGMP query message in accordance with a preferred embodiment of the present invention. Beginning at step 1402, and upon determining that an IGMP report message has not been

received from a host interface associated with a multicast stream, in step 1404, the logic sends an IGMP "last member" query message over the host interface using the last host address and the IGMP version associated with the multicast stream, in step 1406. The logic then waits for an IGMP report message from the host interface, in step 1408. If the logic receives an IGMP report message over the host interface (YES in step 1410), then the logic leaves the host interface on the host interface list. If the logic does not receive an IGMP report message over the host interface (NO in step 1410), then the logic removes the host interface from the host interface list associated with the multicast stream, in step 1412. The logic 1400 terminates in step 1499.

FIG. 15 is a logic flow diagram showing exemplary logic 1500 for forwarding a multicast packet in accordance with a preferred embodiment of the present invention. Beginning at step 1502, and upon receiving a multicast packet over an interface, in step 1504, the logic determines the multicast stream associated with the multicast packet, in step 1506, and forwards the multicast packet to all interfaces indicated in the host interface list and/or the router interface list other than the interface over which the multicast packet was received, in step 1508. The logic 1500 terminates in step 1599.

When a cooperating module is reconfigured to operate in a stand-alone mode of operation or otherwise becomes inactive in the stack, all remaining modules must stop forwarding information to the reconfigured module, and the reconfigured module must stop forwarding information to the remaining modules. Therefore, a leave stack event is generated within the reconfigured module, and a module inactive event including a module identifier identifying the reconfigured module is sent to the remaining modules. The leave stack event causes the reconfigured module, and in particular the IGMP snooping/proxy logic (115), to remove all remote modules (stack ports) from all host interface lists and from all router interface lists, and consequently from the forwarding interface list. Upon receiving the module inactive event, each remaining module, and in particular the IGMP snooping/proxy logic (115), removes the reconfigured module (stack port) from all host interface lists and from all router interface lists, and consequently removes the reconfigured module (stack port) from the forwarding interface list (111).

FIG. 16 is a logic flow diagram showing exemplary logic 1600 for removing an

inactive remote module in accordance with a preferred embodiment of the present invention. Beginning at step 1602, and upon receiving a module inactive event identifying an inactive remote module, in step 1604, the logic removes the inactive remote module (stack port) from all host interface lists, in step 1606, and removes the inactive remote module (stack port) from all router interface lists, in step 1608. The logic 1600 terminates in step 1699.

FIG. 17 is a logic flow diagram showing exemplary logic 1700 for entering a stand-alone mode of operation in accordance with a preferred embodiment of the present invention. Beginning at step 1702, and upon receiving a leave stack event, in step 1704, the logic removes all remote modules (stack ports) from all host interface lists, in step 1706, and removes all remote modules (stack ports) from all router interface lists, in step 1708. The logic 1700 terminates in step 1799.

In a preferred embodiment of the present invention, the intermediate device is a modular Ethernet switching device having a plurality of cooperating Ethernet switching modules. However, the present invention is in no way limited to a modular Ethernet switching device. The present invention may be applied more generally in a distributed communication system having a plurality of cooperating modules that are interconnected, for example, via a bus, backplane, or other signaling medium.

In a preferred embodiment of the present invention, predominantly all of the IGMP snooping/proxy logic is implemented as a set of computer program instructions that are stored in a computer readable medium and executed by an embedded microprocessor system within the module. Preferred embodiments of the invention may be implemented in any conventional computer programming language. For example, preferred embodiments may be implemented in a procedural programming language (*e.g.*, "C") or an object oriented programming language (*e.g.*, "C++"). Alternative embodiments of the invention may be implemented using discrete components, integrated circuitry, programmable logic used in conjunction with a programmable logic device such as a Field Programmable Gate Array (FPGA) or microprocessor, or any other means including any combination thereof.

Alternative embodiments of the invention may be implemented as a computer

program product for use with a computer system. Such implementation may include a series of computer instructions fixed either on a tangible medium, such as a computer readable media (*e.g.*, a diskette, CD-ROM, ROM, or fixed disk), or fixed in a computer data signal embodied in a carrier wave that is transmittable to a computer system via a modem or other interface device, such as a communications adapter connected to a network over a medium. The medium may be either a tangible medium (*e.g.*, optical or analog communications lines) or a medium implemented with wireless techniques (*e.g.*, microwave, infrared or other transmission techniques). The series of computer instructions embodies all or part of the functionality previously described herein with respect to the system. Those skilled in the art should appreciate that such computer instructions can be written in a number of programming languages for use with many computer architectures or operating systems. Furthermore, such instructions may be stored in any memory device, such as semiconductor, magnetic, optical or other memory devices, and may be transmitted using any communications technology, such as optical, infrared, microwave, or other transmission technologies. It is expected that such a computer program product may be distributed as a removable medium with accompanying printed or electronic documentation (*e.g.*, shrink wrapped software), preloaded with a computer system (*e.g.*, on system ROM or fixed disk), or distributed from a server or electronic bulletin board over the network (*e.g.*, the Internet or World Wide Web).

The present invention may be embodied in other specific forms without departing from the essence or essential characteristics. The described embodiments are to be considered in all respects only as illustrative and not restrictive.

We claim:

1. A method for forwarding multicast packets by a module in a communication system having a plurality of cooperating modules, each module including at least one network interface for supporting directly connected multicast devices and an inter-module communication interface for communicating with remote modules, the method comprising:

determining that a remote module is associated with a multicast stream;

receiving a multicast packet associated with the multicast stream from a network interface; and

forwarding the multicast packet to the remote module over the inter-module communication interface.

2. The method of claim 1, wherein determining that the remote module is associated with the multicast stream comprises:

receiving an IGMP message from the remote module over the inter-module communication interface.

3. The method of claim 1, further comprising:

maintaining a forwarding interface list indicating all network interfaces and remote modules that are associated with the multicast stream;

determining that the multicast stream is supported on at least one network interface of the remote module; and

adding the remote module to the forwarding interface list upon determining that the multicast stream is supported on at least one network interface of the remote module.

4. The method of claim 3, further comprising:

determining that the remote module is not a cooperating module; and

removing the remote module from the forwarding interface list.

5. The method of claim 3, further comprising:

reconfiguring the module to operate in a stand-alone mode of operation; and

removing all remote modules from the forwarding interface list.

6. The method of claim 1, further comprising:

determining that a network interface is associated with the multicast stream;

receiving a multicast packet associated with the multicast stream from the remote module over the inter-module communication interface; and

forwarding the multicast packet to the network interface.

7. The method of claim 1, further comprising:

receiving an IGMP message from the network interface; and

forwarding the IGMP message to all remote modules over the inter-module communication interface.

8. A method for sending IGMP messages by a device in a communication system, the method comprising:

receiving a first IGMP message including a multicast device address; and

sending a second IGMP message using the multicast device address from the first IGMP message.

9. The method of claim 8, wherein the multicast device address is a multicast host address, and wherein the second IGMP message is an IGMP report message.

10. The method of claim 8, wherein the multicast device address is a multicast router address, and wherein the second IGMP message is an IGMP query message.

11. A method for sending IGMP messages by a device in a communication system, the device including an interface, the method comprising:

receiving a plurality of IGMP messages over the interface, wherein each IGMP message received over the interface is one of an IGMP version 1 message and an IGMP version 2 message; and

sending an IGMP version 1 message over the interface if at least one of the plurality of IGMP messages received over the interface is an IGMP version 1 message.

12. The method of claim 11, further comprising:

sending an IGMP version 2 message over the interface if and only if each of the plurality of IGMP messages received over the interface is an IGMP version 2 message.

13. A module for forwarding multicast packets in a communication system having a plurality of cooperating modules, the module comprising:

at least one network interface for supporting directly connected multicast devices;

an inter-module communication interface for communicating with remote modules; and

switching logic operably coupled to receive a first multicast packet from a network interface, determine a first multicast stream for the first multicast packet, and forward the first multicast packet to all remote modules that are associated with the first multicast stream over the inter-module communication interface.

14. The module of claim 13, further comprising a forwarding interface list indicating all remote modules that are associated with the first multicast stream.

15. The module of claim 14, wherein the switching logic is operably coupled to forward the first multicast packet to all remote modules that are indicated in the forwarding interface list.

16. The module of claim 13, further comprising snooping logic operably coupled to determine remote modules that are associated with the first multicast stream based upon IGMP messages received from the remote modules.

17. The module of claim 16, wherein the snooping logic is operably coupled to receive an IGMP message from a remote module and add the remote module to a forwarding interface list.

18. The module of claim 13, wherein the switching logic is operably coupled to receive a second multicast packet from a remote module over the inter-module communication interface, determine a second multicast stream for the multicast packet, and forward the second multicast packet to all network interfaces that are associated with the second multicast stream.

19. The module of claim 18, further comprising a forwarding interface list indicating all network interfaces that are associated with the second multicast stream.

20. The module of claim 19, wherein the switching logic is operably coupled to forward the second multicast packet to all network interfaces that are indicated in the forwarding interface list.

21. The module of claim 18, further comprising snooping logic operably coupled to determine network interfaces that are associated with the second multicast stream based upon IGMP messages received from the network interfaces.

22. The module of claim 21, wherein the snooping logic is operably coupled to receive an IGMP message from a network interface and add the network interface to a forwarding interface list.

23. A device for sending IGMP messages in a communication system, the device comprising:

receiving logic operably coupled to receive a first IGMP message including a multicast device address; and

sending logic operably coupled to send a second IGMP message using the multicast device address from the first IGMP message.

24. The device of claim 23, wherein the multicast device address is a multicast host address, and wherein the second IGMP message is an IGMP report message.

25. The device of claim 23, wherein the multicast device address is a multicast router address, and wherein the second IGMP message is an IGMP query message.

26. A device for sending IGMP messages in a communication system, the device comprising:

receiving logic operably coupled to receive a plurality of IGMP messages over an interface, wherein each IGMP message received over the interface is one of an IGMP version 1 message and an IGMP version 2 message; and

sending logic operably coupled to send an IGMP version 1 message over the interface if at least one of the plurality of IGMP messages received over the interface is an IGMP version 1 message.

27. The device of claim 26, wherein the sending logic is operably coupled to send an IGMP version 2 message over the interface if and only if each of the plurality of IGMP messages received over the interface is an IGMP version 2 message.

28. A program product comprising a computer readable medium having embodied therein a computer program for forwarding multicast packets by a module in a communication system having a plurality of cooperating modules, the module including at least one network interface for supporting directly connected multicast devices and an inter-module communication interface for communicating with remote modules, the computer program comprising:

receiving logic programmed to receive an IGMP message from a remote module over the inter-module communication interface; and

snooping logic programmed to determine that the remote module supports a multicast stream based upon the IGMP message received from the received IGMP message.

29. The program product of claim 28, wherein the snooping logic is programmed to add the remote module to a forwarding interface list that indicates all network interfaces and remote modules associated with the multicast stream.

30. The program product of claim 28, further comprising:

switching logic programmed to receive multicast packets associated with the multicast stream from a network interface and forward the multicast packets to the remote module over the inter-module communication interface

31. A program product comprising a computer readable medium having embodied therein a computer program for sending IGMP messages by a device in a communication system, the computer program comprising:

receiving logic programmed to receive a first IGMP message including a multicast device address; and

sending logic programmed to send a second IGMP message using the multicast device address from the first IGMP message.

32. The program product of claim 31, wherein the multicast device address is a multicast host address, and wherein the second IGMP message is an IGMP report message.

33. The program product of claim 31, wherein the multicast device address is a multicast router address, and wherein the second IGMP message is an IGMP query message.

34. A program product comprising a computer readable medium having embodied therein a computer program for sending IGMP messages by a device in a communication system, the computer program comprising:

receiving logic programmed to receive a plurality of IGMP messages over an interface, wherein each IGMP message received over the interface is one of an IGMP version 1 message and an IGMP version 2 message; and

sending logic programmed to send an IGMP version 1 message over the interface if at least one of the plurality of IGMP messages received over the interface is an IGMP version 1 message.

35. The program product of claim 34, wherein the sending logic is programmed to send an IGMP version 2 message over the interface if and only if each of the plurality of IGMP messages received over the interface is an IGMP version 2 message.

36. A distributed multicast switching system comprising a plurality of cooperating modules, each module including at least one network interface for supporting directly connected multicast devices and an inter-module communication interface for communicating with remote modules, wherein each module is operably coupled to forward IGMP messages received from the at least one network interface to all remote modules over the inter-module communication interface, determine all remote modules that are associated with a multicast stream based upon IGMP messages received from the remote modules, and forward multicast packets received from the at least one network interface to all remote modules that are associated with the multicast stream.

ABSTRACT OF THE DISCLOSURE

A technique for multicast switching in a distributed communication system having a plurality of cooperating modules enables a module to forward multicast packets associated with a multicast stream without using a centralized module or control logic by determining all network interfaces and remote modules associated with the multicast stream and forwarding multicast packets to only those network interfaces and remote modules associated with the multicast stream. IGMP snooping may be used to determine the network interfaces and remote modules associated with the multicast stream, and may also be used to determine host addresses, router addresses, and an IGMP version for each network interface and for each remote module. In order to generate IGMP messages, the module learns a multicast device address from received IGMP messages and uses the multicast device address to send the IGMP messages. The module decides whether to send IGMP version 1 messages or IGMP version 2 messages on a particular interface based upon the IGMP version of IGMP messages received over the interface.

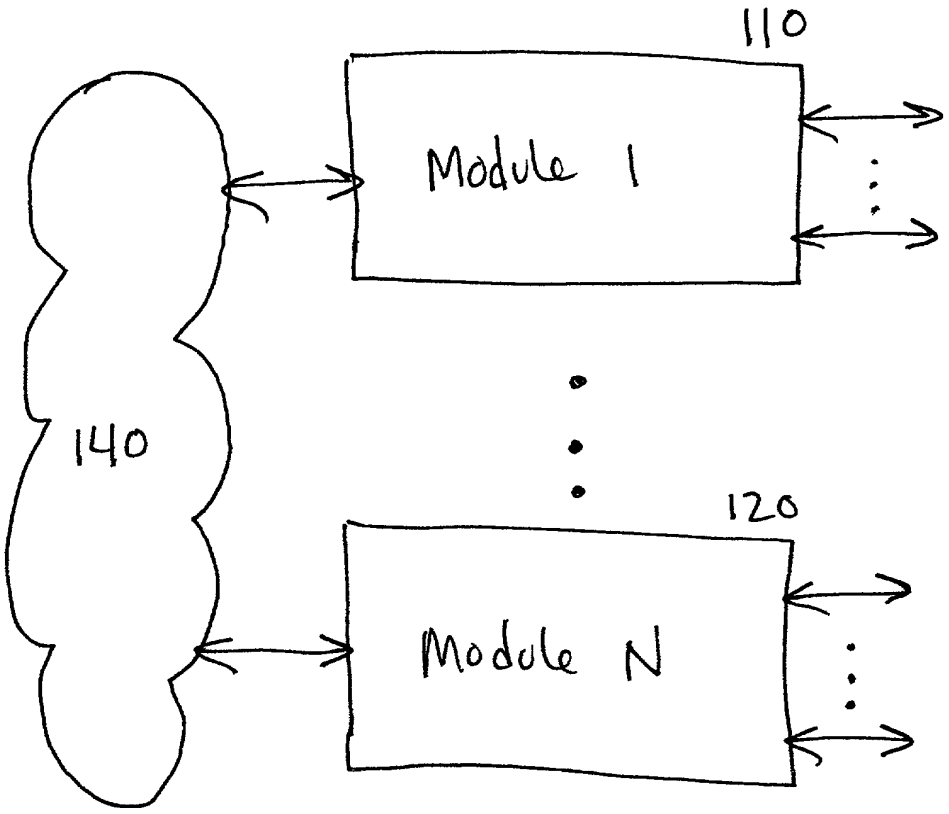


FIG. 1 100

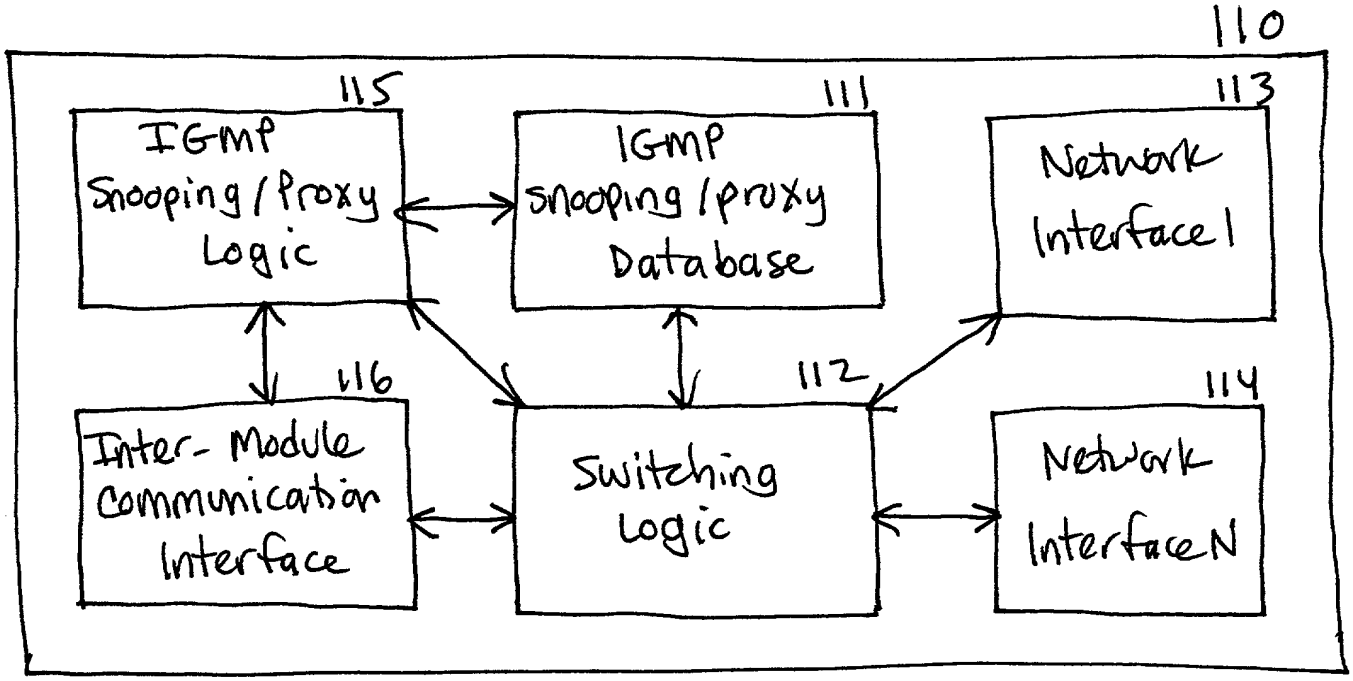

110
115
IGMP Snooping / Proxy Logic
111
IGMP snooping / proxy Database
113
Network Interface 1
116
Inter-Module Communication Interface
112
Switching Logic
114
Network Interface N

FIG. 2

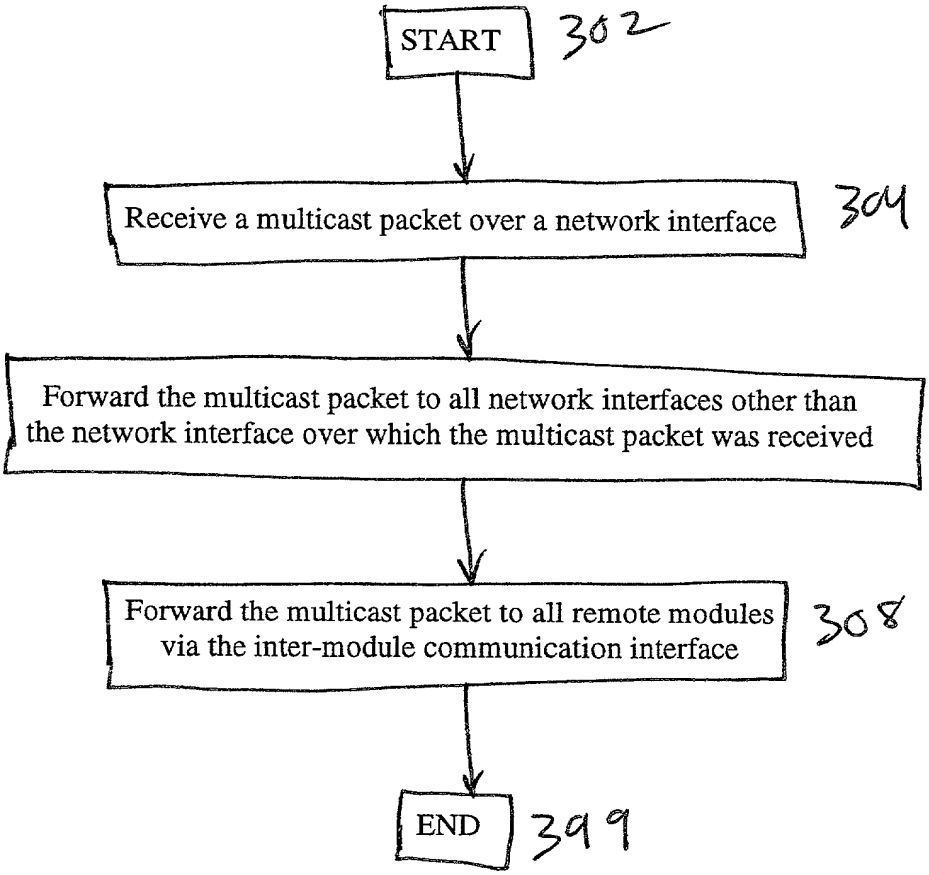

START
302
Receive a multicast packet over a network interface
304
Forward the multicast packet to all network interfaces other than the network interface over which the multicast packet was received
306
Forward the multicast packet to all remote modules via the inter-module communication interface
308
END
399

FIG. 3 300

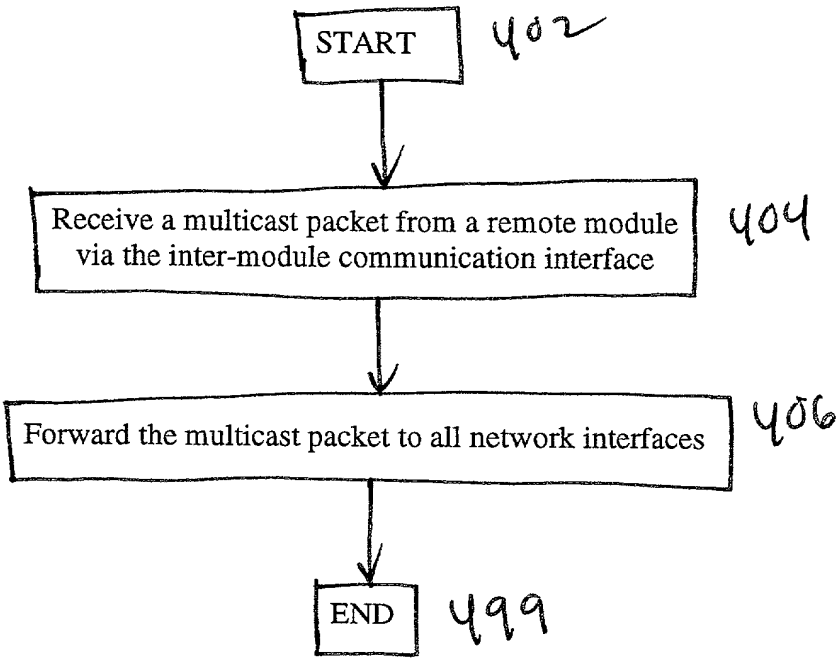

START
402
Receive a multicast packet from a remote module via the inter-module communication interface
404
Forward the multicast packet to all network interfaces
406
END
499

FIG. 4 400

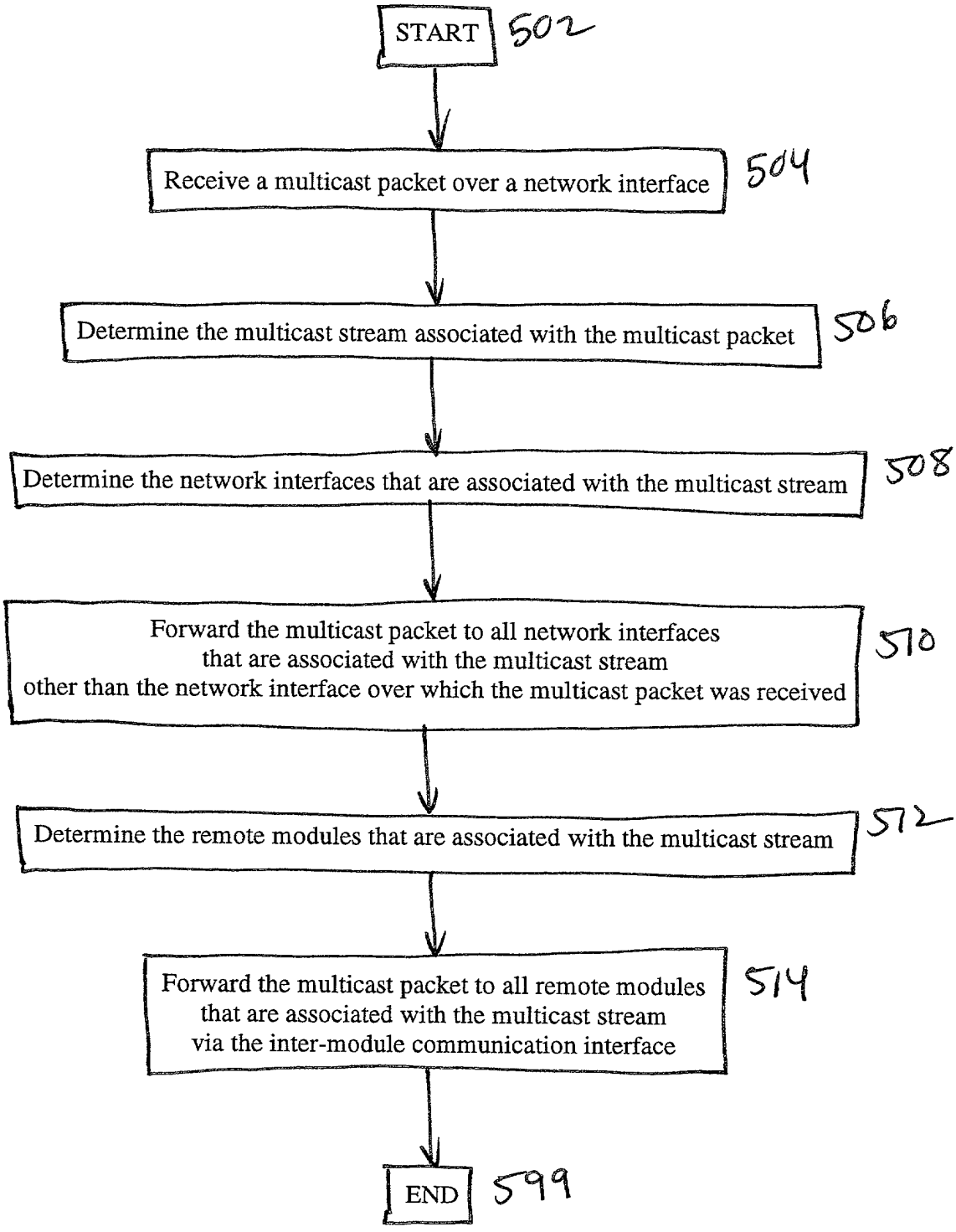

START
502
Receive a multicast packet over a network interface
504
Determine the multicast stream associated with the multicast packet
506
Determine the network interfaces that are associated with the multicast stream
508
Forward the multicast packet to all network interfaces
that are associated with the multicast stream
other than the network interface over which the multicast packet was received
510
Determine the remote modules that are associated with the multicast stream
512
Forward the multicast packet to all remote modules
that are associated with the multicast stream
via the inter-module communication interface
514
END
599

FIG. 5 500

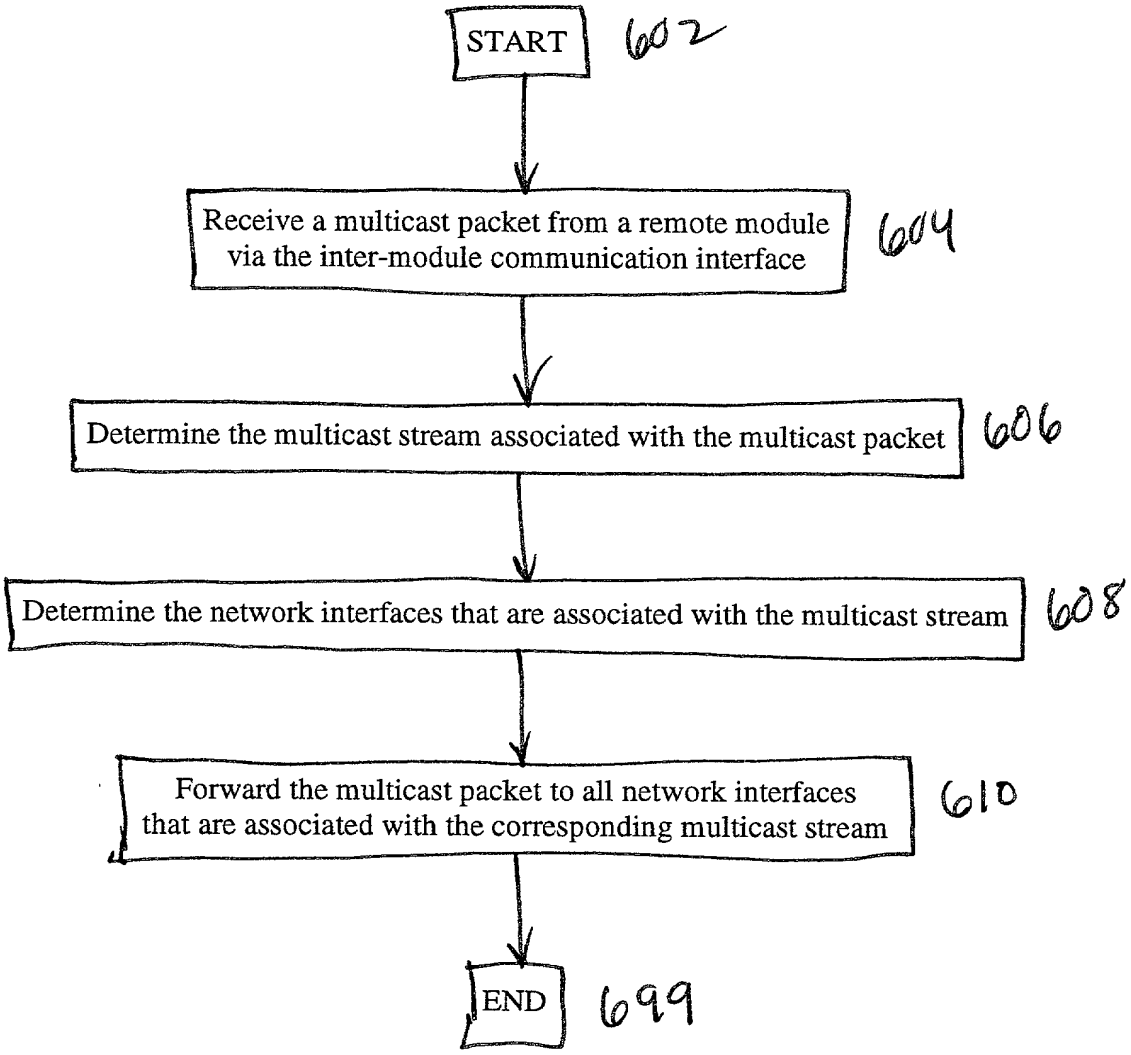

START
602
Receive a multicast packet from a remote module via the inter-module communication interface
604
Determine the multicast stream associated with the multicast packet
606
Determine the network interfaces that are associated with the multicast stream
608
Forward the multicast packet to all network interfaces that are associated with the corresponding multicast stream
610
END
699

FIG. 6 600

Last Host Address — 702

Last Router Address — 704

Host Interface List — 706

Router Interface List — 708

IGMP Version List — 710

FIG. 7

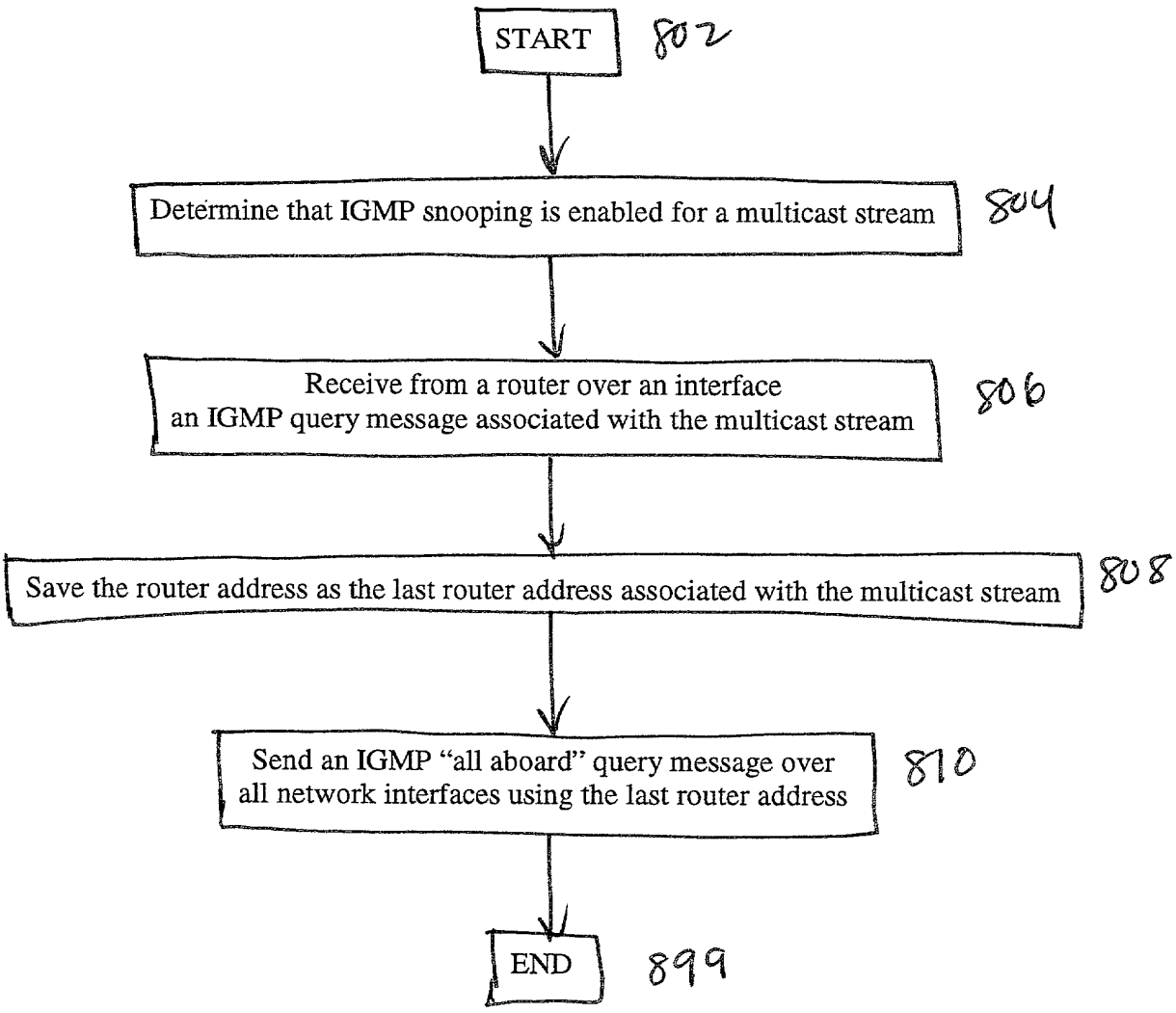

START
802
Determine that IGMP snooping is enabled for a multicast stream
804
Receive from a router over an interface
an IGMP query message associated with the multicast stream
806
Save the router address as the last router address associated with the multicast stream
808
Send an IGMP "all aboard" query message over
all network interfaces using the last router address
810
END
899

FIG. 8 800

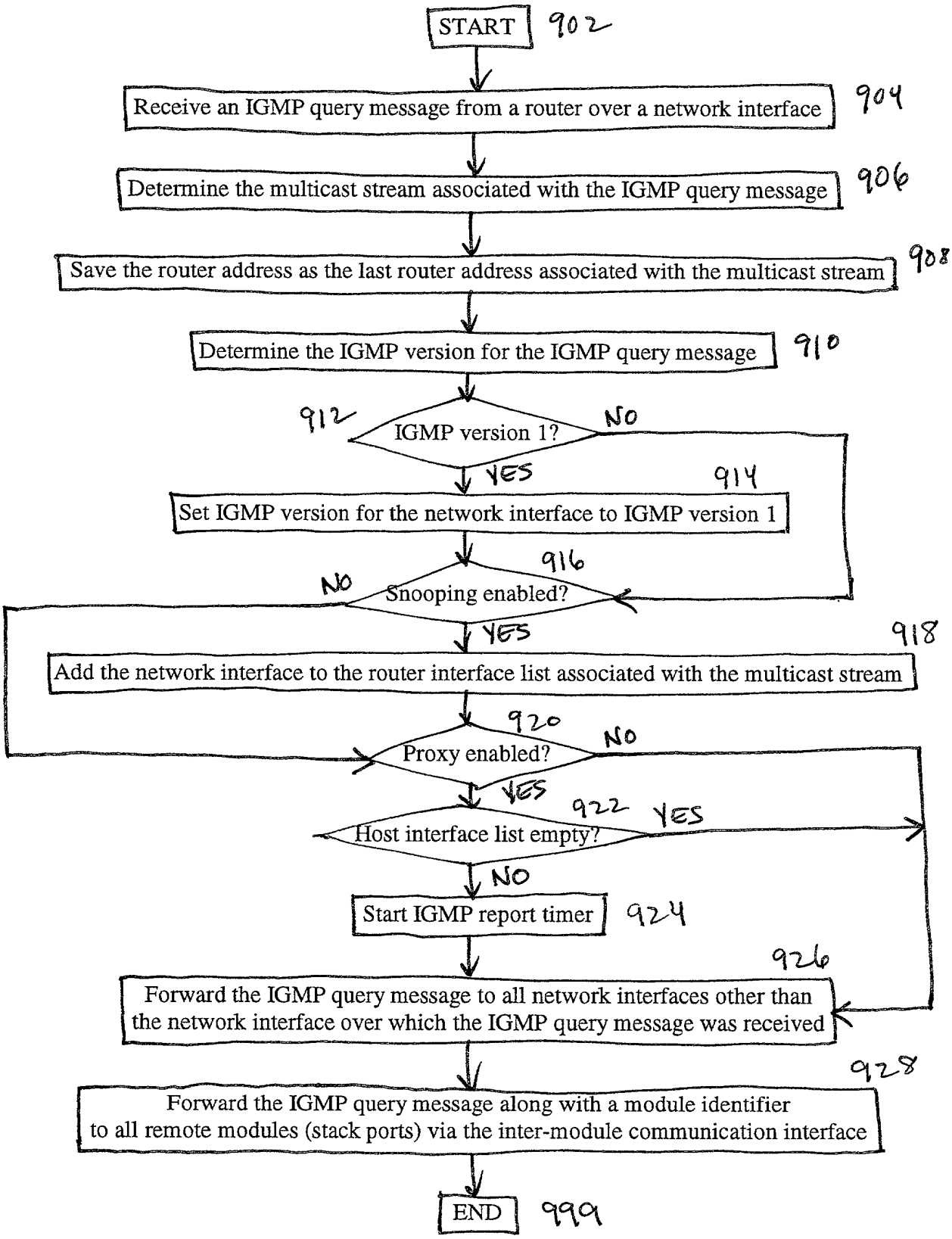

START
902
Receive an IGMP query message from a router over a network interface
904
Determine the multicast stream associated with the IGMP query message
906
Save the router address as the last router address associated with the multicast stream
908
Determine the IGMP version for the IGMP query message
910
912
IGMP version 1?
NO
YES
914
Set IGMP version for the network interface to IGMP version 1
916
NO
Snooping enabled?
YES
918
Add the network interface to the router interface list associated with the multicast stream
920
Proxy enabled?
NO
YES
922
YES
Host interface list empty?
NO
Start IGMP report timer
924
926
Forward the IGMP query message to all network interfaces other than the network interface over which the IGMP query message was received
928
Forward the IGMP query message along with a module identifier to all remote modules (stack ports) via the inter-module communication interface
END
999

FIG. 9 900

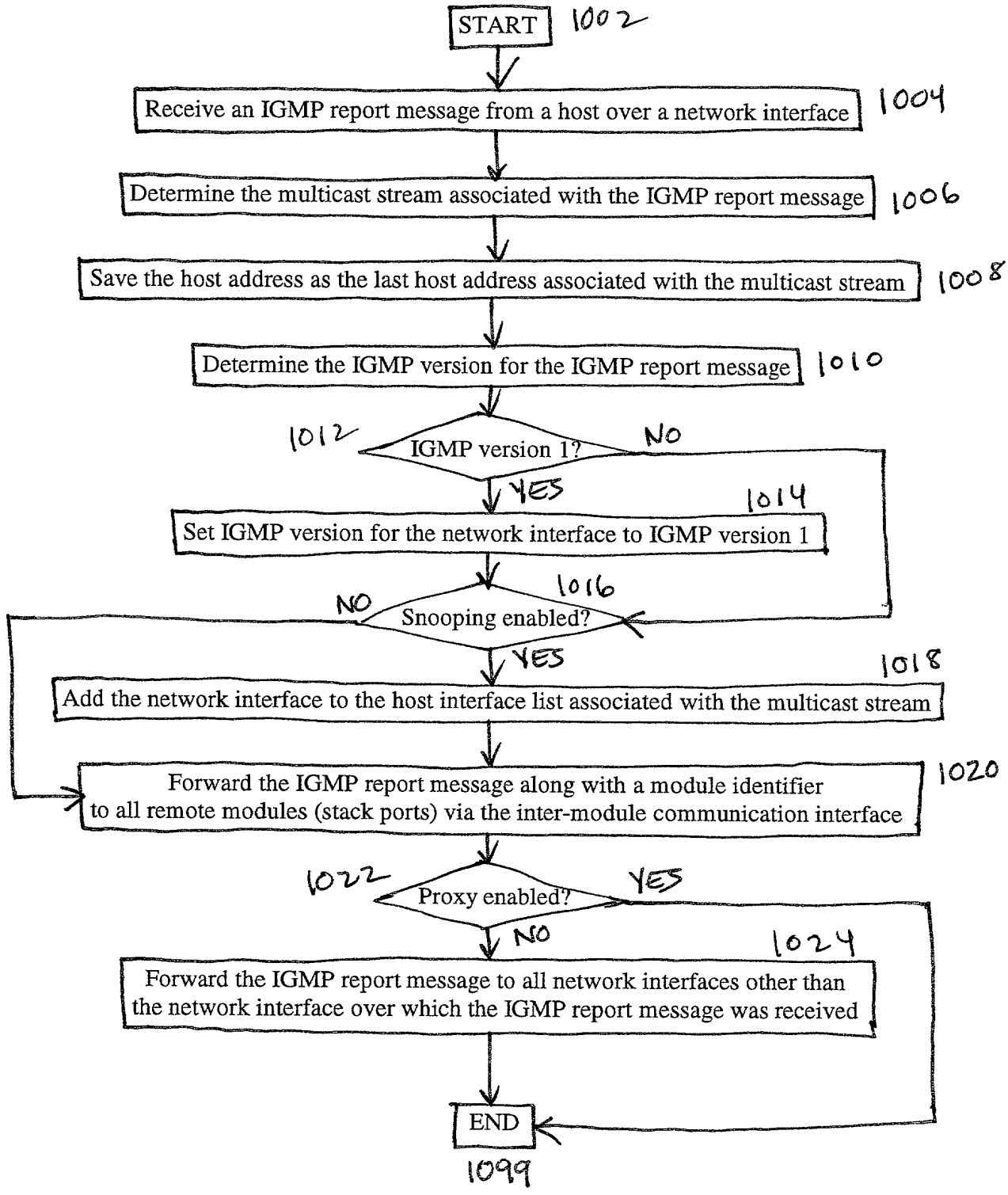

START
1002
Receive an IGMP report message from a host over a network interface
1004
Determine the multicast stream associated with the IGMP report message
1006
Save the host address as the last host address associated with the multicast stream
1008
Determine the IGMP version for the IGMP report message
1010
1012
IGMP version 1?
NO
YES
1014
Set IGMP version for the network interface to IGMP version 1
1016
NO
Snooping enabled?
YES
1018
Add the network interface to the host interface list associated with the multicast stream
Forward the IGMP report message along with a module identifier
to all remote modules (stack ports) via the inter-module communication interface
1020
1022
Proxy enabled?
YES
NO
1024
Forward the IGMP report message to all network interfaces other than
the network interface over which the IGMP report message was received
END
1099

FIG 10 1000

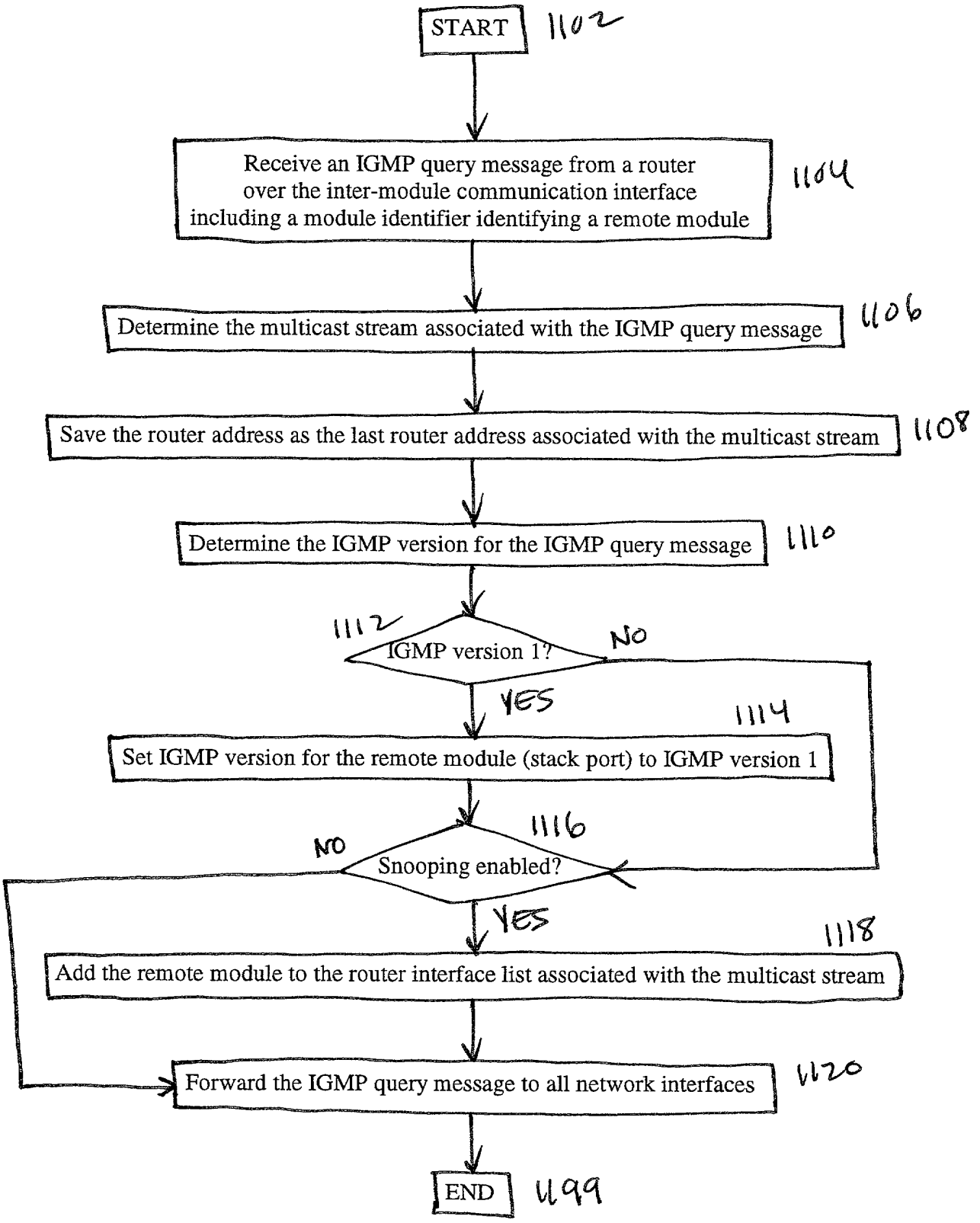

START
1102
Receive an IGMP query message from a router over the inter-module communication interface including a module identifier identifying a remote module
1104
Determine the multicast stream associated with the IGMP query message
1106
Save the router address as the last router address associated with the multicast stream
1108
Determine the IGMP version for the IGMP query message
1110
1112
IGMP version 1?
NO
YES
1114
Set IGMP version for the remote module (stack port) to IGMP version 1
1116
NO
Snooping enabled?
YES
1118
Add the remote module to the router interface list associated with the multicast stream
Forward the IGMP query message to all network interfaces
1120
END
1199

FIG. 11 1100

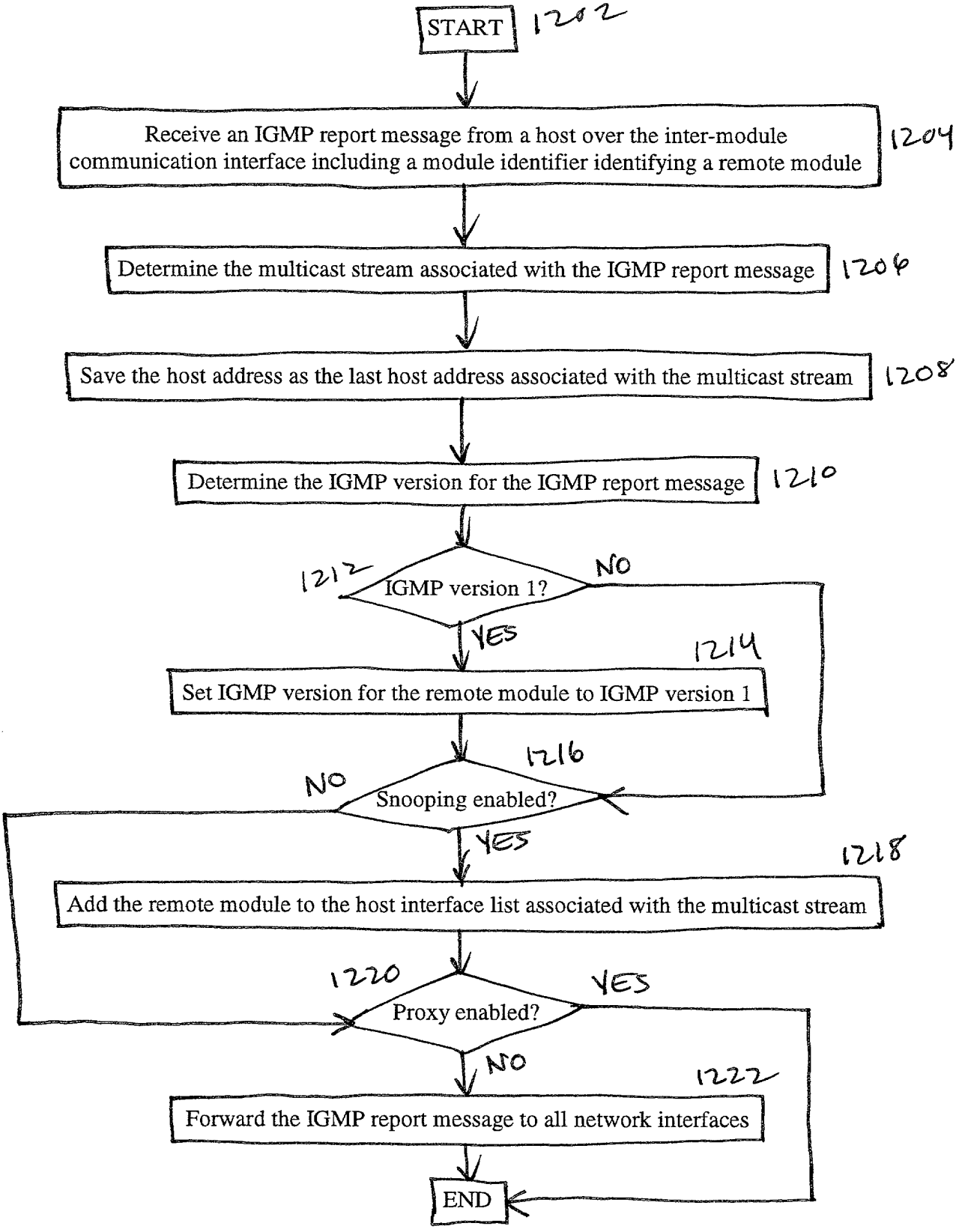

START
1202
Receive an IGMP report message from a host over the inter-module communication interface including a module identifier identifying a remote module
1204
Determine the multicast stream associated with the IGMP report message
1206
Save the host address as the last host address associated with the multicast stream
1208
Determine the IGMP version for the IGMP report message
1210
1212
IGMP version 1?
NO
YES
1214
Set IGMP version for the remote module to IGMP version 1
1216
NO
Snooping enabled?
YES
1218
Add the remote module to the host interface list associated with the multicast stream
1220
YES
Proxy enabled?
NO
1222
Forward the IGMP report message to all network interfaces
END

FIG. 12 1200

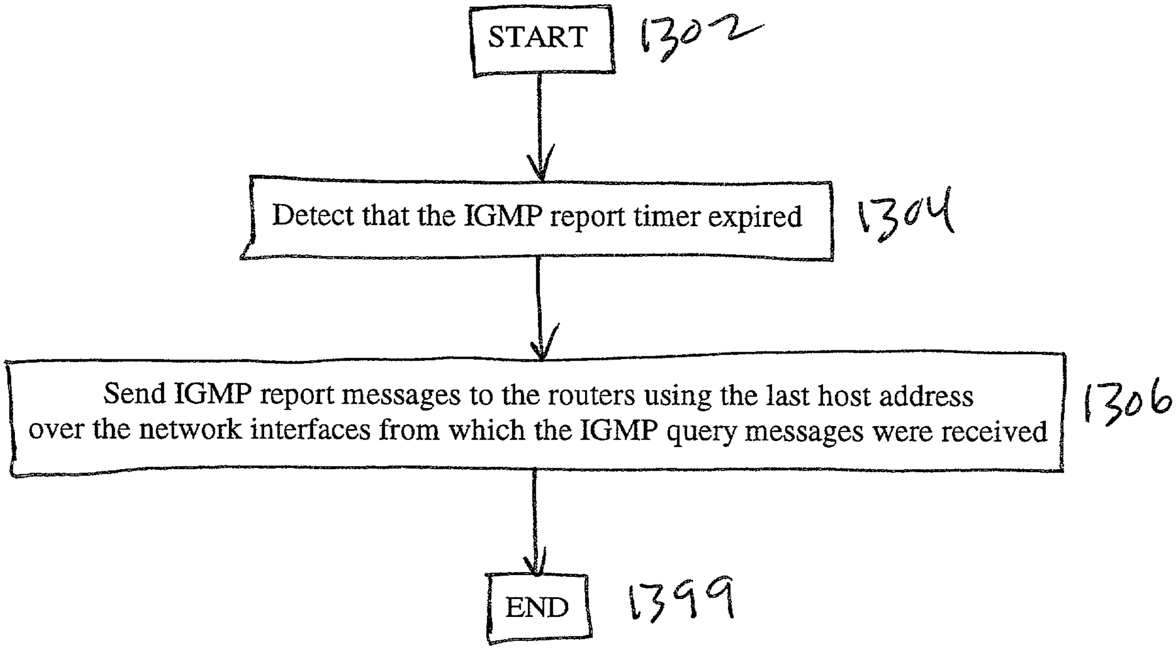

START
1302
Detect that the IGMP report timer expired
1304
Send IGMP report messages to the routers using the last host address over the network interfaces from which the IGMP query messages were received
1306
END
1399

FIG. 13 1300

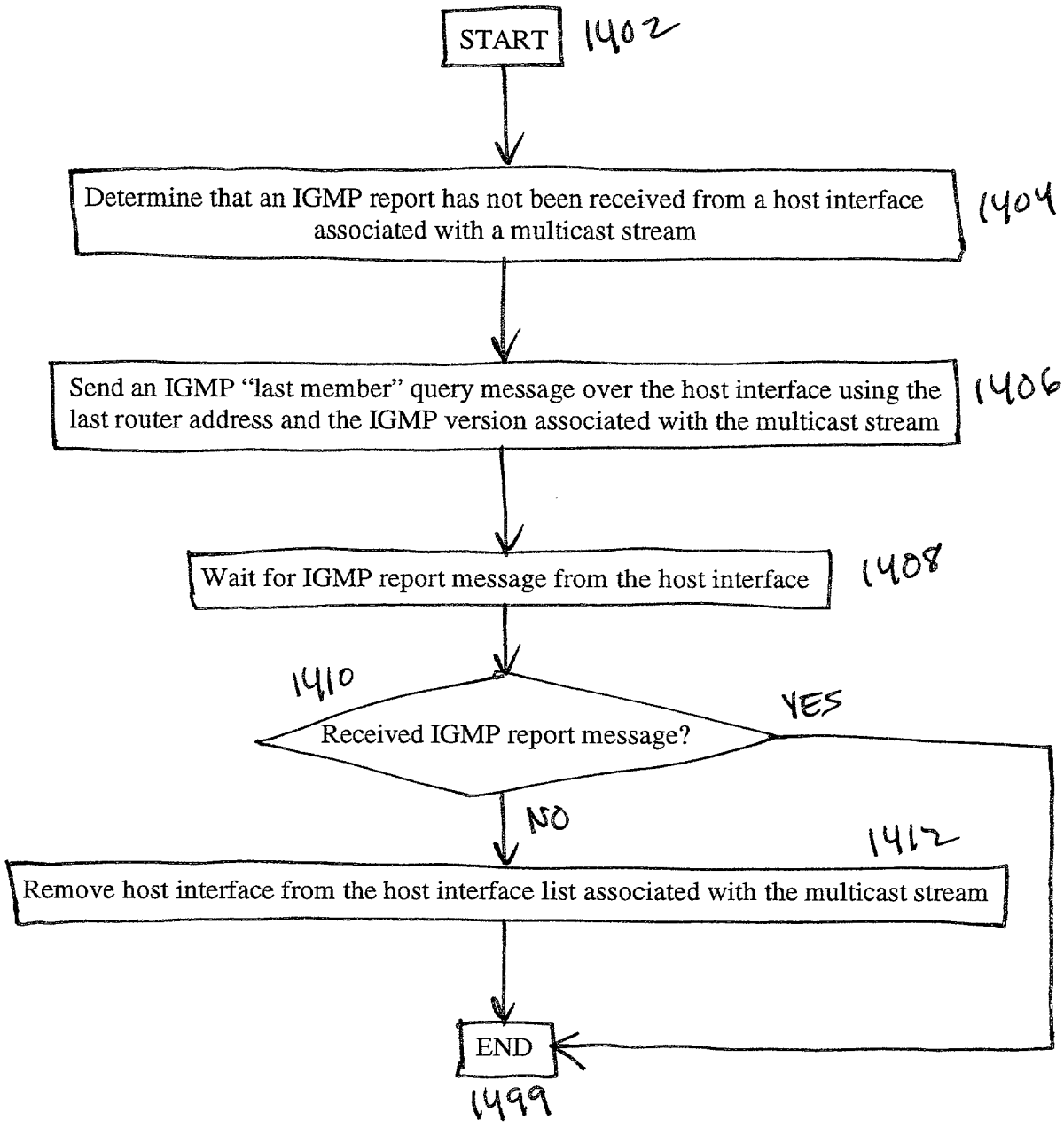

START
1402
Determine that an IGMP report has not been received from a host interface associated with a multicast stream
1404
Send an IGMP "last member" query message over the host interface using the last router address and the IGMP version associated with the multicast stream
1406
Wait for IGMP report message from the host interface
1408
1410
Received IGMP report message?
YES
NO
1412
Remove host interface from the host interface list associated with the multicast stream
END
1499

FIG. 14 1400

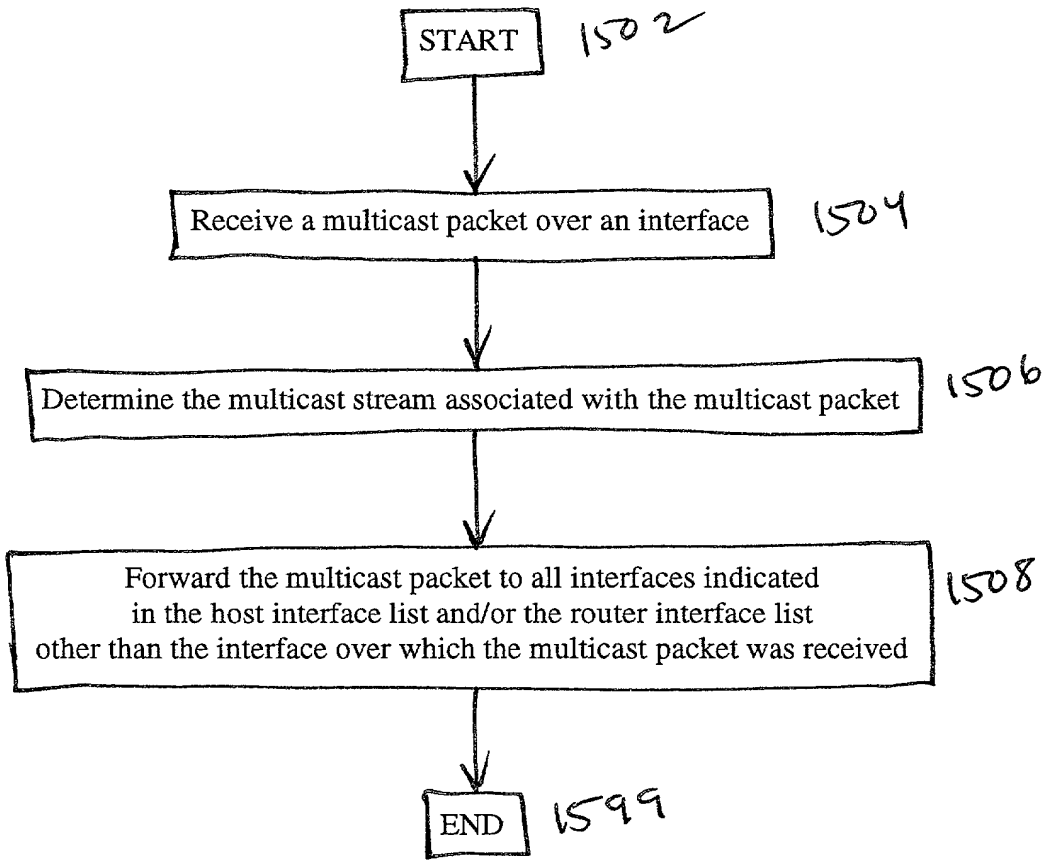

START
1502
Receive a multicast packet over an interface
1504
Determine the multicast stream associated with the multicast packet
1506
Forward the multicast packet to all interfaces indicated in the host interface list and/or the router interface list other than the interface over which the multicast packet was received
1508
END
1599

FIG. 15 1500

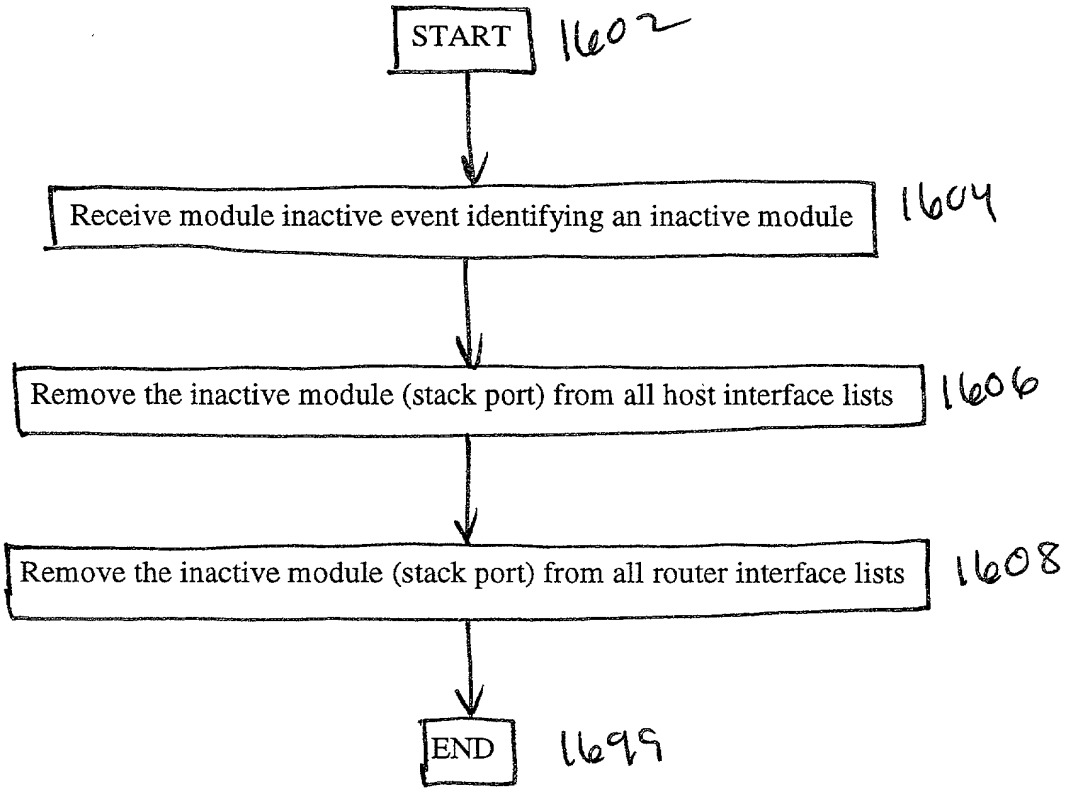

START
1602
Receive module inactive event identifying an inactive module
1604
Remove the inactive module (stack port) from all host interface lists
1606
Remove the inactive module (stack port) from all router interface lists
1608
END
1699

FIG. 16 1600

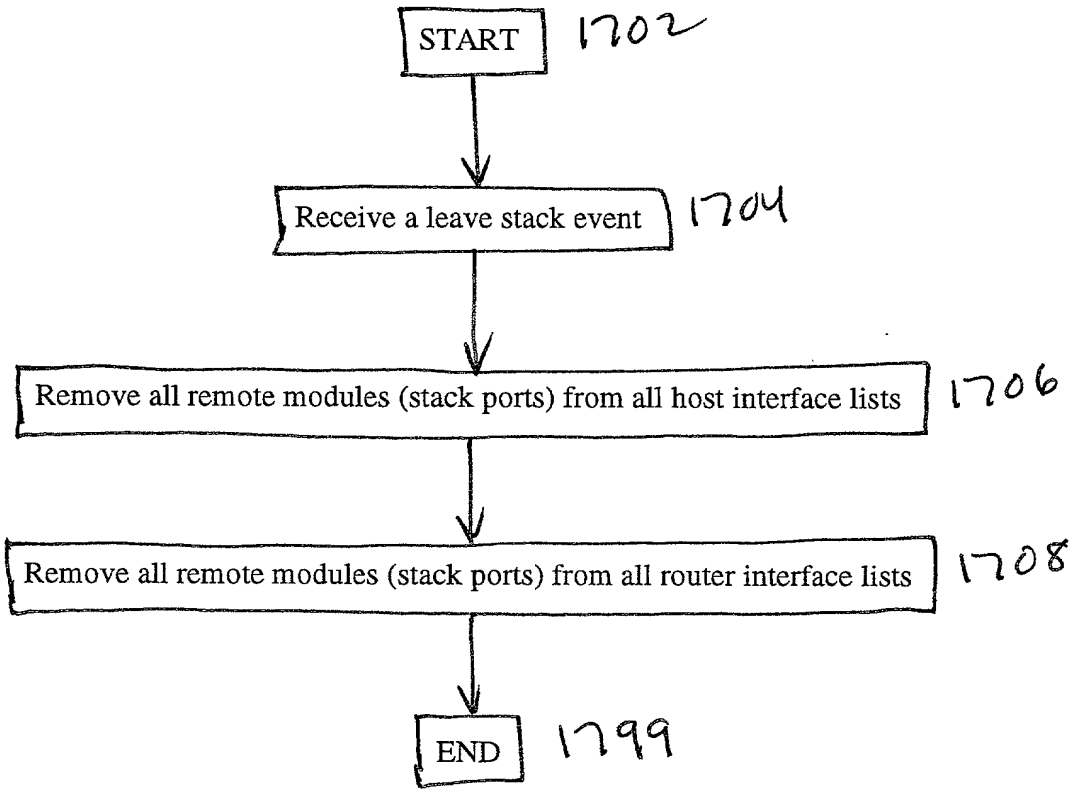

START
1702
Receive a leave stack event
1704
Remove all remote modules (stack ports) from all host interface lists
1706
Remove all remote modules (stack ports) from all router interface lists
1708
END
1799

FIG 17 1700

Docket No.
2204/193

Declaration and Power of Attorney For Patent Application

English Language Declaration

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

MULTICAST SWITCHING IN A DISTRIBUTED COMMUNICATION SYSTEM

the specification of which

(check one)

☒ is attached hereto.

☐ was filed on _____ as United States Application No. or PCT International Application Number _____

and was amended on _____

(if applicable)

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose to the United States Patent and Trademark Office all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, Section 119(a)-(d) or Section 365(b) of any foreign application(s) for patent or inventor's certificate, or Section 365(a) of any PCT International application which designated at least one country other than the United States, listed below and have also identified below, by checking the box, any foreign application for patent or inventor's certificate or PCT International application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application(s)			Priority Not Claimed
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐

I hereby claim the benefit under 35 U.S.C. Section 119(e) of any United States provisional application(s) listed below:

(Application Serial No.)	(Filing Date)
(Application Serial No.)	(Filing Date)
(Application Serial No.)	(Filing Date)

I hereby claim the benefit under 35 U. S. C. Section 120 of any United States application(s), or Section 365(c) of any PCT International application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of 35 U.S.C. Section 112, I acknowledge the duty to disclose to the United States Patent and Trademark Office all information known to me to be material to patentability as defined in Title 37, C. F. R., Section 1.56 which became available between the filing date of the prior application and the national or PCT International filing date of this application:

(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith. *(list name and registration number)*

Bruce D. Sunstein	Reg. No. 27,234	Jay Sandvos	Reg. No. 43,900
Robert M. Asher	Reg. No. 30,445	Karen A. Buchanan	Reg. No. 37,790
Timothy M. Murphy	Reg. No. 33,198	Sonia K. Guterman	Reg. No. 44,729
Steven G. Saunders	Reg. No. 36,265	Keith J. Wood	Reg. No. P45,235
Harriet M. Strimpel	Reg. No. 37,008	Mary M. Steubing	Reg. No. 37,946
Samuel J. Petuchowski	Reg. No. 37,910	Christopher J. Cianciolo	Reg. No. 42,417
Jeffrey T. Klayman	Reg. No. 39,250	Lindsay J. McGuinness	Reg. No. 38,549
John J. Stickevers	Reg. No. 39,387		
Herbert A. Newborn	Reg. No. 42,031		
Elizabeth P. Morano	Reg. No. 42,904		
Jean M. Tibbetts	Reg. No. 43,193		

Send Correspondence to: Jeffrey T. Klayman
Bromberg & Sunstein LLP
125 Summer Street
Boston, MA 02110

Direct Telephone Calls to: *(name and telephone number)*
Jeffrey T. Klayman at (617) 443-9292

Full name of sole or first inventor **Da-Hai Ding**
Sole or first inventor's signature — Date
Residence **10 Baker Avenue, Lexington, MA 02421**
Citizenship **U.S.A.**
Post Office Address **Same as residence**

Full name of second inventor, if any
Second inventor's signature — Date
Residence
Citizenship
Post Office Address